

Article

# Surrogate-Assisted Symbolic Time-Series Discretization Using Multi-Breakpoints and a Multi-Objective Evolutionary Algorithm

Aldo Márquez-Grajales <sup>1,\*</sup>, Efrén Mezura-Montes <sup>1</sup>, Héctor-Gabriel Acosta-Mesa <sup>1</sup>  
and Fernando Salas-Martínez <sup>2</sup>

<sup>1</sup> Artificial Intelligence Research Institute, University of Veracruz, Campus Sur, Calle Paseo Lote II, Sección Segunda 112, Nuevo Xalapa, Veracruz 91097, Mexico; emezura@uv.mx (E.M.-M.); heacosta@uv.mx (H.-G.A.-M.)

<sup>2</sup> El Colegio de Veracruz, Carrillo Puerto 26, Colonia Centro, Xalapa, Veracruz 91000, Mexico; fersantz@gmail.com

\* Correspondence: li.aldomg@gmail.com

**Abstract:** The enhanced multi-objective symbolic discretization for time series (eMODiTS) method employs a flexible discretization scheme using different value cuts for each non-equal time interval, which incurs a high computational cost for evaluating each objective function. It is essential to mention that each solution found by eMODiTS is a different-sized vector. Previous work was performed where surrogate models were implemented to reduce the computational cost to solve this problem. However, low-fidelity approximations were obtained concerning the original model. Consequently, our main objective is to propose an improvement to this work, modifying the updating process of the surrogate models to minimize their disadvantages. This improvement was evaluated based on classification, predictive power, and computational cost, comparing it against the original model and ten discretization methods reported in the literature. The results suggest that the proposal achieves a higher fidelity to the original model than previous work. It also achieved a computational cost reduction rate between 15% and 80% concerning the original model. Finally, the classification error of our proposal is similar to eMODiTS and maintains its behavior compared to the other discretization methods.

**Keywords:** surrogate models; time series representation; symbolic representation; multi-objective optimization



**Citation:** Márquez-Grajales, A.; Mezura-Montes, E.; Acosta-Mesa, H.-G.; Salas-Martínez, F. Surrogate-Assisted Symbolic Time-Series Discretization Using Multi-Breakpoints and a Multi-Objective Evolutionary Algorithm. *Math. Comput. Appl.* **2024**, *29*, 78. <https://doi.org/10.3390/mca29050078>

Received: 31 May 2024

Revised: 31 August 2024

Accepted: 9 September 2024

Published: 11 September 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

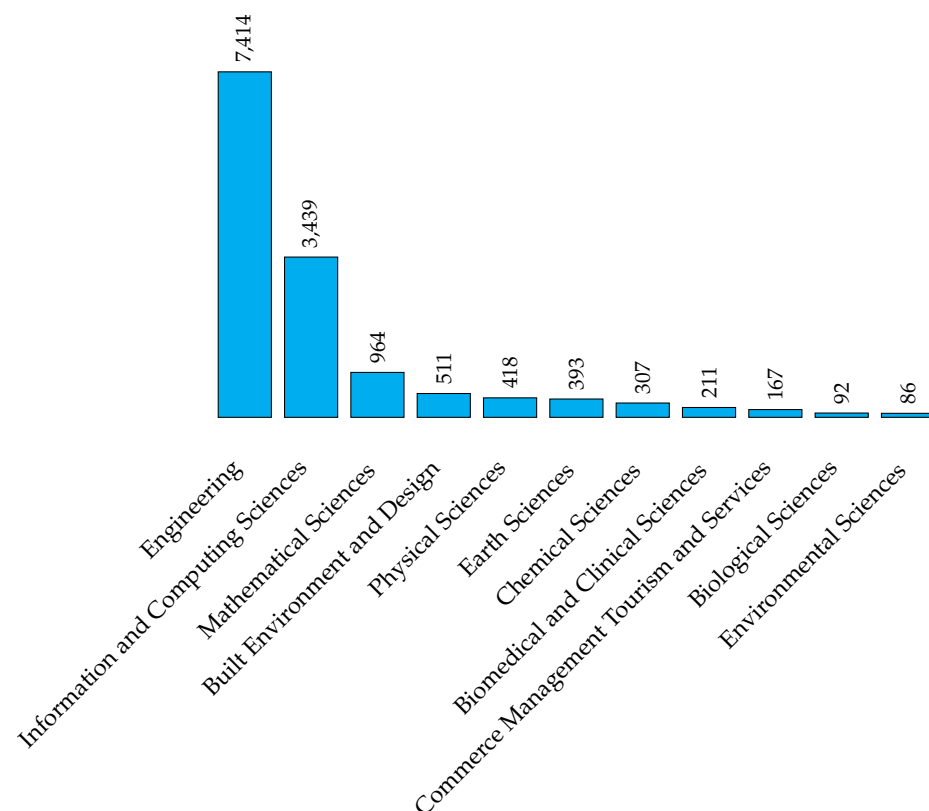
Time series discretization transforms continuous values into discrete ones [1]. Symbolic discretization is one of the most widely used approaches for transforming time series due to its ability to exploit data richness and its lower bounding properties, among others [2,3]. On the other hand, the Symbolic Aggregate Approximation (SAX) is the most widely used symbolic discretization method due to its easy conception, implementation, and low computational time [2,3].

SAX employs the well-known Piecewise Aggregate Approximation (PAA). PAA reduces the dimensionality of the time series by averaging the values falling on one of the equal-sized time intervals defined at the beginning of the SAX procedure. Finally, each average value is mapped to a set of breakpoints computed based on the Gaussian distribution to assign the corresponding symbol. Despite SAX's advantages, SAX has been criticized for its Gaussian distribution assumption and its inevitable information loss when the dimensionality of the data is reduced [4–7].

As a consequence, several approaches have emerged to improve the limitations of SAX, such as Extended SAX (ESAX) [3,8], adaptive ( $\alpha$ SAX) [9], Random Shifting-based SAX

(rSAX) [10], one-dimensional SAX (1D-SAX) [11], Hexadecimal Aggregate approximation (HAX) [12], Point Aggregate approximation (PAX) [12], season SAX (sSAX) [13], trend SAX (tSAX) [13], probabilistic SAX (pSAX) [14], clustering SAX (cSAX) [14], Symbolic Aggregate approximation based on Kmeans (SAX-Kmeans) [15], and Extended Symbolic Aggregate approximation based on Kmeans (ESAX-Kmeans) [15]. These methods use a local search to find the discretization scheme. However, some methods focus on implementing a global search for this task; for example, the Evolutionary Programming (EP) approach [16], Harmony SAX (HSAX) [17], Genetic Algorithm SAX (GASAX) [18], Differential Evolution SAX (DESAX) [19], and eMODiTS [20]. The main drawback of these techniques is their high computational cost due to the increased number of operations needed to find a solution.

One way to reduce this cost is the use of surrogate models. Surrogate models reduce the processing time in most complex optimization problems [21]. The areas where surrogate models are applied are diverse. According to the literature, engineering is the area with the most applications of these models [22–29], followed by information and computing sciences [30–33], mathematical sciences [34,35], and built environment and design [36,37], among others; see Figure 1.

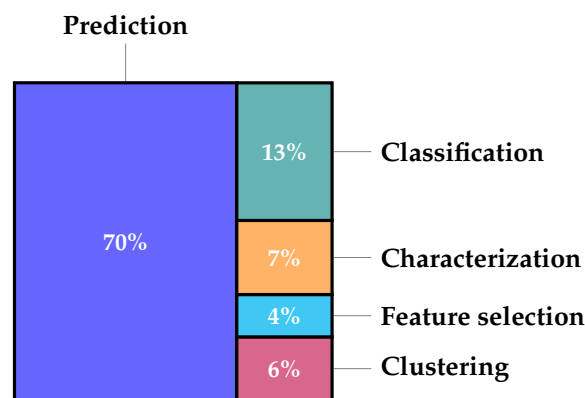


**Figure 1.** Amount of publications on surrogate models to date. This bibliographic analysis was elaborated using Dimensions AI [38] and the keywords *surrogate model optimization* or *approximation model optimization*.

Moreover, surrogate models are employed for machine learning tasks. In this sense, an analysis of the use of surrogate models in each machine learning task is presented in Figure 2. This figure shows that the *prediction* task is the most recurring data mining task where an approximation model is used instead of the original prediction model. However, for tasks such as classification, the number of publications is less than the number of publications examining predictions, resulting in an opportunity niche for researchers. Mostly, the surrogate model research applied to time series classification is focused on hyperparameter optimization [39], deep learning [40], and neuroevolution [41]. Nevertheless, as far as the

state-of-the-art has been reviewed, surrogate models have been scarcely implemented in time series discretization.

In this context, ref. [42] is one of the few works in which surrogate models were implemented to minimize the computational cost by identifying an appropriate discretization scheme for temporal data. The researchers applied surrogate models to eMODiTS [20]. eMODiTS is a temporal data mining technique that discretizes time series using a unique set of breakpoint values for each time segment and three objective functions. The dataset is discretized before computing the value of each objective function for an individual. Hence, this process implies a high computational cost to evaluate each objective function, representing an essential disadvantage of the method. Therefore, surrogate models were employed to approximate the values of the three objective functions. As the individual representation size of eMODiTS varies across instances, k-nearest neighbor (KNN) was employed as a surrogate model. Moreover, the Pareto front was evaluated in the original models every  $N$ th generation and added to the training set to update the surrogate models. Nevertheless, this resulted in a lessened update of the surrogate model, thereby reducing the model's fidelity.



**Figure 2.** Marimekko chart searched at <https://www.dimensions.ai/> using the keywords “Surrogate model optimization” OR “approximation model optimization” and “clustering”, “classification”, “prediction”, “associative analysis”, “characterization”, and “feature selection”.

Consequently, our primary motivation for this research is to extend the findings of [42], proposing a novel surrogate model update process, being the particular contribution of this work. The proposed methodology evaluates the Pareto front using the original objective functions at each generation (individual-based strategy) and the current population at regular intervals (generation-based strategy). Additionally, each time the generation-based update is applied, the Pareto front (evaluated on the original functions) will be incorporated into the training set. This proposal aims to increase the number of times the model is updated to increase model fidelity.

Therefore, the objectives of this research are described below:

1. To increase the number of evaluations conducted on the original problem functions, thereby improving the fidelity of the surrogate models.
2. To maintain the accuracy of the classification task achieved by the original model (eMODiTS).
3. To analyze the surrogate model behavior compared with SAX-based discretization methods to verify whether the proposal maintains, improves, or worsens by incorporating these models regarding the well-known discretization approaches.

The organization of this document is presented as follows. First, Section 2 describes the materials and methods used in this research and the implemented methodology. On the other hand, Section 3 presents the experiments performed to reach the objectives introduced in Section 1 and the discussion of the results. Finally, Section 4 describes the conclusions based on the results presented in the previous section.

## 2. Materials and Methods

This section describes the materials (data set) and methods used to understand and reproduce our proposal.

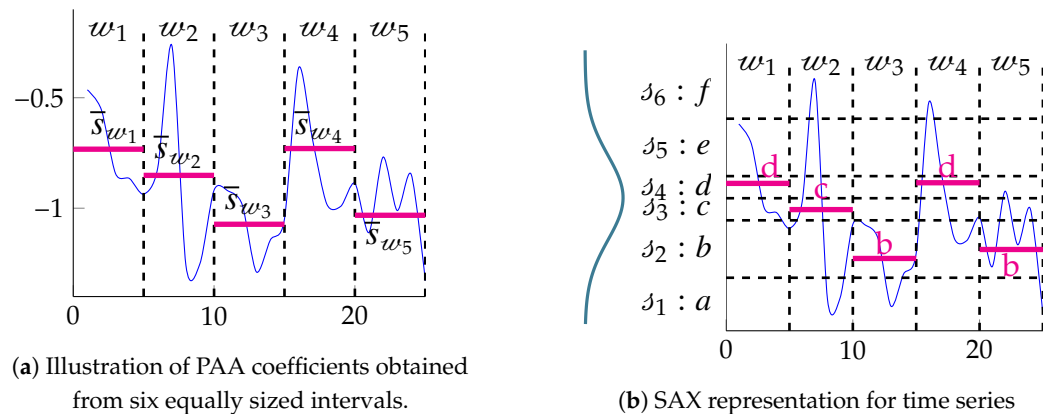
### 2.1. Symbolic Aggregate appRoXimation (SAX)

SAX is the most famous symbolic time series discretization method. SAX is based on the dimensionality reduction method called PAA.

PAA partitions the time series into equal  $p$  intervals  $\mathcal{W}$ , averaging the time series values within each interval. Each average value is called the *PAA coefficient*.

**Definition 1.** Let  $\mathcal{T}_i$  be a time series,  $\tau_i$  be a time series value at time  $t$ , and  $\mathcal{W} = \{w_1, w_2, w_3, \dots, w_{|\mathcal{W}|}\}$  be a set of equal-sized intervals ( $w_i = [w_i, w_{i+1}), i \neq |\mathcal{W}|$ ). The **PAA coefficient**  $\bar{s}$  of the interval  $w_i$  is computed as  $\bar{s}_{w_i} = \frac{1}{\eta} \sum_{t \in w_i} s_t$ , where  $\eta$  is the number of time series values within the interval  $w_i$ .

The calculation of the PAA coefficients is visually represented in Figure 3a. In this particular case, there are six intervals of equal size, denoted as  $|\mathcal{W}| = 6$ . For each interval  $w_i$ , the coefficient  $\bar{s}_{w_i}$  is determined based on the values within that interval.



**Figure 3.** Representations for PAA and SAX algorithms. The final string  $dcbdb$  is obtained by mapping each PAA coefficient  $\bar{s}_{w_i}$  to a symbol  $s_j$ .

SAX is a method that transforms time series into reduced-symbolic discretization using the PAA approach. The transformation process involves mapping each PAA coefficient (referred to as *word segments*) to symbols. It is achieved by defining a set of intervals, known as *breakpoints* or *alphabet cuts*, in the value space based on a normal distribution. Each interval in the set represents a symbol or character, denoted as  $s_j$ . The PAA coefficient that falls within a specific interval is then replaced by its corresponding symbol. The resulting collection of symbols is called *string*.

**Definition 2.** Let  $\mathcal{A} = \{(\alpha_1, s_1), (\alpha_2, s_2), (\alpha_3, s_3), \dots, (\alpha_{|\mathcal{A}|}, s_{|\mathcal{A}|})\}$  be a set of intervals or breakpoints in the value spaces  $\alpha_j = [\alpha_j, \alpha_{j+1}), j \neq |\mathcal{A}|$  with their corresponding symbol  $s_j$ . A **string** ( $\mathcal{S}$ ) is formed as  $\mathcal{S} = \{s_j | \bar{s}_{w_i} \in \alpha_j, \forall i \in \{1, 2, \dots, |\mathcal{W}|\} \text{ and } \forall j \in \{1, 2, \dots, |\mathcal{A}|\}\}$ .

Figure 3b shows the SAX representation of the PAA coefficients shown in Figure 3a. In this example, the breakpoints are  $|\mathcal{A}| = 6$ . Each PAA coefficient  $s_{w_i}$  is replaced by a corresponding symbol according to the interval in which it falls. The resulting string for this example is  $dcbdb$ . It is worth mentioning that SAX requires two crucial initial parameters: the number of word segments ( $|\mathcal{W}|$ ) and the number of breakpoints ( $|\mathcal{A}|$ ). Both parameters impact the overall behavior of the approach. Normalizing the time series before discretization is also necessary, as the breakpoints are generated based on this distribution.

## 2.2. Multi-Objective Optimization Problem (MOOP)

MOOP involves simultaneously optimizing functions via maximizing, minimizing, or both. Furthermore, each possible solution results in a trade-off in each function [43]. Therefore, MOOP is expressed in Definition 3.

**Definition 3.** Let  $\vec{x} = [x_1, x_2, \dots, x_{|\vec{x}|}]$  be the vector of decision variables for a problem, where  $|\vec{x}|$  is the number of variables,  $\vec{F}$  is the set of objective functions to be optimized,  $g_i(\vec{x})$  and  $h_e(\vec{x})$  are inequality and equality constrained functions, respectively,  $I$  is the number of inequality functions,  $E$  is the number of equality functions, and  $L$  and  $U$  are the lower and upper domain values of the decision variables space, respectively. The **MOOP** can be stated as shown in Equation (1) [43–45].

$$\begin{aligned}
 & \text{Minimize} \\
 & \quad \vec{F} = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_{|\vec{F}|}(\vec{x})]^T \\
 & \text{Subject to :} \\
 & \quad g_i(\vec{x}) \leq 0, \text{ for } i = \{1, 2, \dots, I\} \\
 & \quad h_e(\vec{x}) = 0, \text{ for } e = \{1, 2, \dots, E\} \\
 & \quad \vec{x} \in [L, U]
 \end{aligned} \tag{1}$$

Instead of finding a unique solution, MOOP aims to find a set of feasible solutions [46]. Nevertheless, locating this set of feasible solutions poses a significant challenge. The Pareto-optimality approach is widely utilized in MOOP due to its ability to produce competitive outcomes [47,48]. Furthermore, Pareto introduced a definition of dominance to determine the superior solution from a given set of solutions [49]. This form of dominance is referred to as *Pareto dominance* and is defined in the following manner.

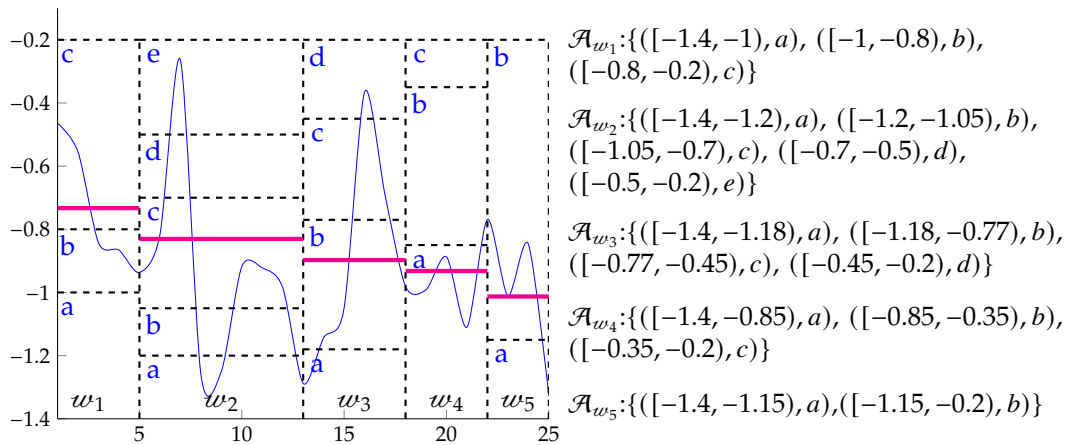
**Definition 4.** Let  $f \in \vec{F}$  be an objective function and  $\vec{x}_1$  and  $\vec{x}_2$  be two feasible solutions.  $\vec{x}_1$  dominates ( $<$ )  $\vec{x}_2$  based on **Pareto dominance**, if and only if  $f_i(\vec{x}_1) \leq f_i(\vec{x}_2) \forall i \in \{1, 2, \dots, |\vec{F}|\}$  and  $f_j(\vec{x}_1) < f_j(\vec{x}_2)$  at least in one  $j \in \{1, 2, \dots, |\vec{F}|\}$  for a minimization problem.

This dominance makes it possible to obtain a collection of non-dominant solutions, called the *Pareto set*, and its image is the Pareto front ( $\mathfrak{F}$ ). From this set, a final solution is chosen based on the user's preference.

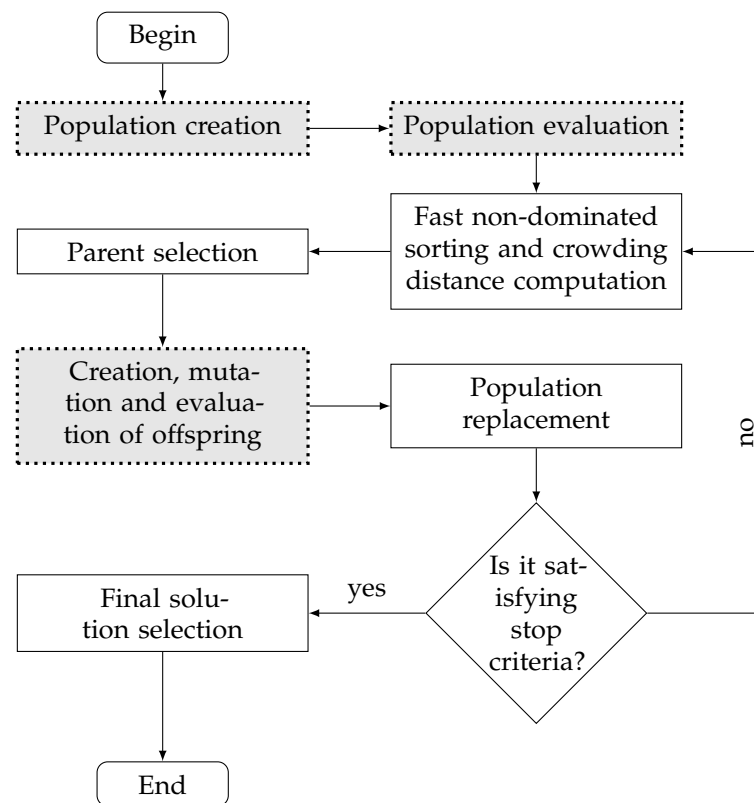
## 2.3. Enhanced Multi-Objective Symbolic Discretization for Time Series (eMODiTS)

eMODiTS is a SAX-based discretization approach [20]. Thus, it employs the PAA technique to reduce dimensionality, word segment cuts, alphabet cuts, and the same symbol assignment process. However, unlike SAX, each word segment includes distinct, random, and non-normal distributed breakpoints. Therefore, it does not assume a normal distribution in time series. A graphical representation of the eMODiTS discretization scheme for the example depicted in Figure 3b is given in Figure 4.

In addition, eMODiTS aims to determine the optimal number of word segments and breakpoints in the discretization process. The eMODiTS approach focuses on optimizing three functions at the same time: entropy, complexity, and information loss. Consequently, the search task to find a suitable discretization scheme employs the well-known Non-dominated Sorting Genetic Algorithm (NSGA-II) [50–52]. Some NSGA-II-specific components were modified to handle the individual representation of eMODiTS. Figure 5 depicts the general discretization procedure employed in eMODiTS. The dotted and light-gray squares represent the modified steps of NSGA-II.



**Figure 4.** eMODiTS scheme discretization approach where each word segment  $w_i$  contains its breakpoint scheme  $\mathcal{A}_{w_i}$ . In this example, the final string is *ccbab*.



**Figure 5.** eMODiTS’s flowchart. Dotted and light gray boxes represent the NSGA-II stages necessary to adapt them to the eMODiTS representation schemes.

### 2.3.1. Population Generation

The population comprises individuals who represent discretization schemes, where each word break includes a distinct breakpoints scheme. The word segments and breakpoints are integer and float value arrays, respectively. As an illustration, Figure 6 displays the vector that represents Figure 4. In this representation, each word segment  $w_i$  is accompanied by its corresponding breakpoints  $\mathcal{A}_{w_i,j}$ . At the start of eMODiTS, these individuals are initialized randomly, taking into account the length and values of a temporal dataset.

$w_1$	$\mathcal{A}_{w_1,1}$	$\mathcal{A}_{w_1,2}$	$w_2$	$\mathcal{A}_{w_2,1}$	$\mathcal{A}_{w_2,2}$	$\mathcal{A}_{w_2,3}$	$\mathcal{A}_{w_2,4}$	...
5	-1	-0.8	13	-1.2	-1.05	-0.7	-0.5	
$w_3$	$\mathcal{A}_{w_3,1}$	$\mathcal{A}_{w_3,2}$	$\mathcal{A}_{w_3,3}$	$w_4$	$\mathcal{A}_{w_4,1}$	$\mathcal{A}_{w_4,2}$	$w_5$	$\mathcal{A}_{w_5,1}$
18	-1.18	-0.77	-0.45	22	-0.85	-0.35	25	-1.15

Figure 6. Representation of an individual in eMODiTS.

### 2.3.2. Evaluation Process

As previously mentioned, the eMODiTS method aims to identify flexible discretization schemes that are based on entropy ( $E$ ), complexity ( $CX$ ), and information loss ( $IL$ ). Each function is specifically targeted in the search for appropriate discretization schemes.  $E$  is employed as a method for estimating classification using a confusion matrix ( $cm$ ) created through the discretized time series ( $\mathcal{S}$ ), where the rows of the confusion matrix are the unique discretized time series, and the columns represent the different classes in a database for each. The data in the matrix indicate the frequency of each class for a particular discretized time series.  $E$  is calculated using Equation (2), where  $P(cm(\mathcal{S}_i, c))$  denotes the probability of a discretized time series being assigned the class  $c$  by  $cm$ . When the measure values are close to zero, it indicates that each time series is assigned to only one class. On the contrary, values close to one suggest that a time series has multiple class labels. Therefore, values close to zero are preferred [16,20].

$$E = \sum_{i=1}^M - \sum_{c=1}^C P(cm(\mathcal{S}_i, c)) \cdot \log_2 P(cm(\mathcal{S}_i, c)) \tag{2}$$

Alternatively,  $CX$  assesses the complexity of the resulting discrete database by considering the number of distinct discretized time series. This measure is used to determine the level of difficulty for classifiers in comprehending the discrete dataset. A value close to zero indicates only a few instances in the discrete dataset, making the classification task more manageable. In contrast, a value close to one suggests that the number of discretized time series is similar to the number of original time series, resulting in a more complex discriminator model for the classifier. Therefore, values close to zero are preferable.  $CX$  is expressed in Equation (3), where  $|\mathcal{D}|$  represents the number of time series in the original dataset,  $|\mathcal{S}|$  is the number of unique discretized time series, and  $C$  denotes the total number of classes in the problem.

$$CX = \begin{cases} \frac{|\mathcal{D}| - |\mathcal{S}| - C}{|\mathcal{D}| + (C-1)} & |\mathcal{S}| - C < 0 \\ \frac{|\mathcal{S}| - C}{|\mathcal{D}| + (C-1)} & otherwise \end{cases} \tag{3}$$

Finally, the information loss ( $IL$ ) measure is used to evaluate the data lost during dimensionality reduction. The Mean Square Error (MSE) measure is a commonly used metric for this purpose. MSE calculates the similarity among two time series of equal size, so it is necessary to reconstruct the discretized time series to compare it with the original ( $\mathcal{T}_i$ ). Values close to zero indicate that the time series are similar, while values close to one indicate dissimilarity.  $IL$  is calculated using Equation (4), where  $\mathcal{R}$  represents a reconstructed time series,  $|\mathcal{R}|$  is the number of reconstructed time series, and  $n$  is the length of the reconstructed time series. Both series are scaled to intervals  $[0, 1]$  for a fair comparison.

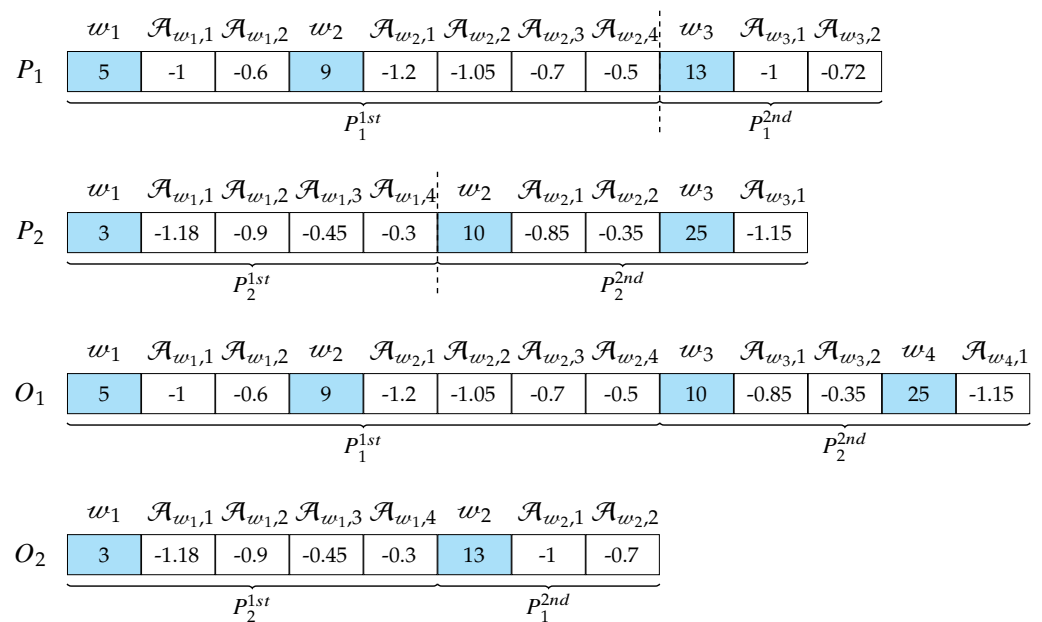
$$IL = \frac{1}{|\mathcal{R}|} \sum_{i=1}^{|\mathcal{R}|} MSE(\mathcal{R}_i, \mathcal{F}_i) \tag{4}$$

$$MSE(\mathcal{R}_i, \mathcal{F}_i) = \frac{\sum (\tau_i - \tau_j)^2}{n - 1}, \tau_i \in \mathcal{R}_i, \tau_j \in \mathcal{F}_i$$

### 2.3.3. Offspring Creation and Mutation

As previously stated, the individuals in eMODiTS consist of varying-sized vectors. Therefore, the crossover operator was modified to generate novel individuals by combining two solutions given this condition. The recombination operator used in this study was the commonly used one-point crossover. In this operator, a random cut is made for both parents, and the resulting segments are then combined [53–57]. However, in this particular case, a different cut is randomly applied for each parent, considering the parents’ sizes. Subsequently, each section of the parents is merged as follows. In the first child ( $O_1$ ), the first part of the first parent ( $P_1^{1st}$ ) corresponds to its first part, while the second part of the second parent ( $P_2^{2nd}$ ) is copied to its last part. For the second child ( $O_2$ ), the first part of the second parent ( $P_2^{1st}$ ) becomes its first part, and the second part of the first parent ( $P_1^{2nd}$ ) fills its second part. Figure 7 illustrates the adapted one-point crossover operator.

In contrast, the uniform mutation operator is utilized, selecting individual positions randomly to modify based on a mutation percentage defined by the user [55,58].



**Figure 7.** Crossover operator based on the one-point approach. The dashed line represents the cuts performed by each parent.

### 2.3.4. Population Replacement

The last process of eMODiTS consists of replacing the current population based on the union of the parent and offspring sets. Fast, non-dominated sorting is applied to obtain the ranking of the parent and offspring set. Subsequently, the next-generation population is filled with the first front of the joined set. If the new population ends with a size smaller than the current population, the second front is selected for the filling procedure. If the updated population size remains less than the current population, the third position is chosen, and so on.



### 2.3.5. Preference Handling

The eMODiTS search mechanism utilizes NSGA-II, which results in a collection of non-dominated solutions (Pareto front) being generated upon completion of the search process rather than a singular solution. Hence, it is essential to have a preference selection mechanism in place to select a single solution from the Pareto front. The preference handling employed in eMODiTS relies on the misclassification rate achieved for each solution on the Pareto front by applying the decision tree classifier to the training set. The final discretization scheme is chosen based on the solution that achieves the lowest misclassification rate. It is important to note that the test set is used solely to demonstrate the final classification performance and not to select the final discretization scheme.

### 2.4. Surrogate-Assisted Multi-Objective Symbolic Discretization for Time Series (sMODiTS)

Due to the flexible discretization scheme integrated into eMODiTS, the search space expands, leading to a higher computational cost. The computational cost is primarily associated with evaluating every solution in the three objective functions. Consequently, a modified approach that integrates approximated models is introduced to reduce the computational cost of eMODiTS. This approach is called *Surrogate-assisted Multi-objective Symbolic Discretization for Time Series (sMODiTS)*. A surrogate model is a substitute in optimization scenarios involving computationally intensive problems. It uses an iterative and updating approach to develop an approximate model that is simpler to manage and less computationally demanding compared to the original model [59–62]. Figure 8 illustrates the general procedure for implementing sMODiTS, highlighting the specific instances when the surrogate model needs to be established and revised.

#### 2.4.1. Training Set Creation

The training set for the surrogate models was generated by applying a uniform design of the experiments. The solutions were randomly initialized within the search space following a uniform distribution [21,63–65]. The design is primarily motivated by the varying number of individual decision variables, making applying traditional initial sampling methods more complex.

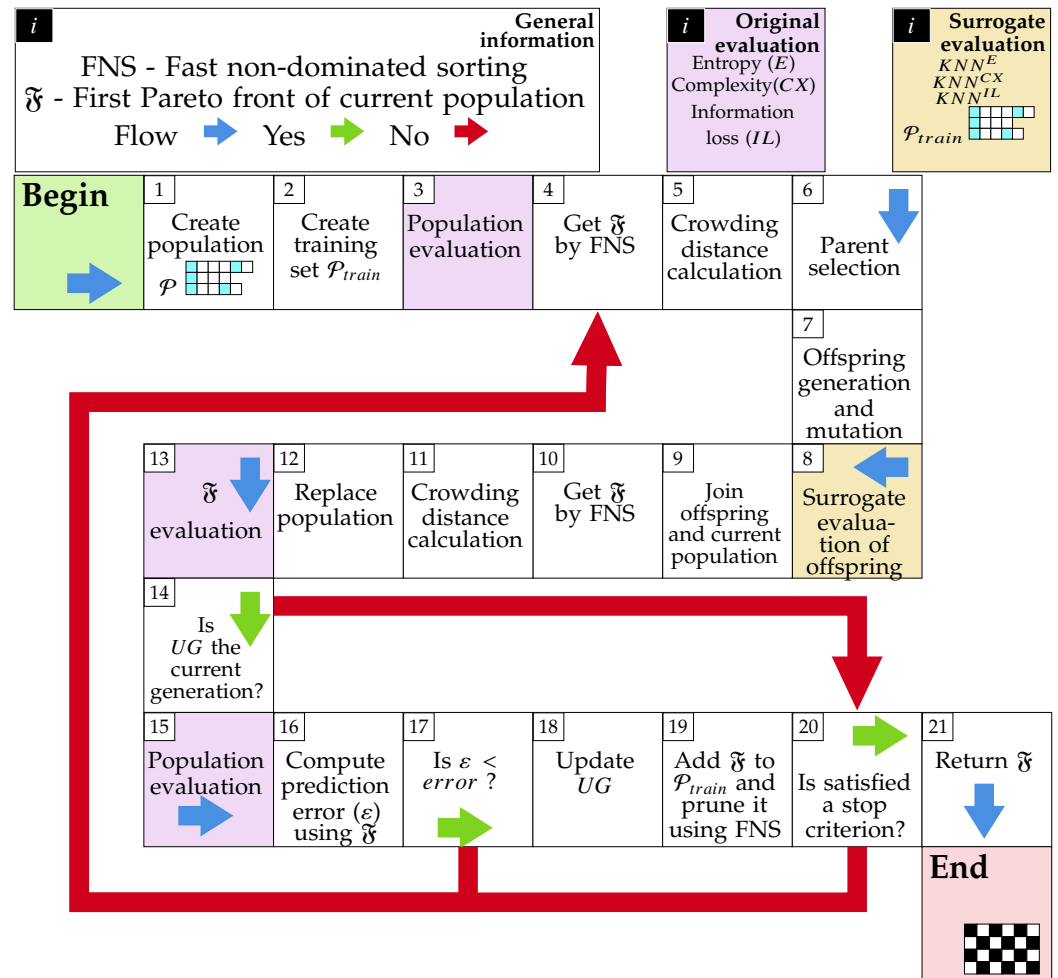
#### 2.4.2. Surrogate Model Creation

Several surrogate models have been used for multi-objective evolutionary optimization [66,67], including the Gaussian process [68–73], Support Vector Machines (SVMs) [29,74,75], Kriging [29,76,77], kernel extreme learning machines [29], Radial Basis Function (RBF) [72,76–79], and artificial neural networks [80–82], among others. These models are usually applied to individuals of the same size. On the other hand, another model used mainly for single-objective optimization is  $k$ -nearest neighbor for regression ( $KNN_R$ ) [61,83,84], which allows for comparison of two unequal-sized vectors by adjusting the distance measure.

$KNN_R$  is based on the original  $KNN$  method. The main difference between them is that  $KNN_R$  uses continuous variables as class labels instead of discrete variables due to the nature of the problems they address. The  $KNN_R$  approach provides the mean class label values of the  $k$  nearest neighbors of a data instance rather than the frequency of the class label. This method is computed using Equation (5), where  $\vec{y}_i$  is a solution belonging to the training set [61].

$$KNN_R(\vec{x}_i) = \frac{\sum_i^k f(\vec{y}_i)}{k} \quad (5)$$

Since  $KNN_R$  is a distance-based classifier and eMODiTS contains solutions of different sizes, an elastic similarity measure was incorporated into  $KNN_R$  to compare two individuals with sMODiTS. Due to the competitive results reported in the state of the art, the selected elastic measure was *Dynamic Time Warping (DTW)*. However, the primary disadvantage of DTW is the computational complexity caused by the exhaustive search for all possible subsequences in the compared vectors.



**Figure 8.** General scheme of sMODiTS. Green arrows indicate paths to follow when the conditions are satisfied. Red arrows indicate paths to follow when conditions are not satisfied. Blue arrows represent the normal flow of the diagram.

### 2.4.3. Surrogate Model Update

Evolution control is a fundamental component of surrogate modeling in optimization tasks. Consequently, sMODiTS implements an individual and generation-based strategy to update each surrogate model. Algorithm 1 demonstrates this process. The individual-based strategy consists of evaluating each individual of the Pareto front in the original objective functions. It is applied to each generation of the evolutionary process. On the other hand, in the generation-based strategy, the current population is evaluated based on the original functions every defined number of generations ( $UG$ ).

Furthermore, in this strategy, Pareto-front individuals are inserted into each training set  $\mathcal{P}_{train}^i$ , and the model is retrained, keeping its size to avoid excessive growth and, therefore, high consumption of computational resources. Consequently, fast, non-dominated sorting was applied to the training set for this process. First, the new training set is filled with non-dominated solutions. Then, if the new training set size is less than the previous set, the following non-dominated solutions are implemented without counting the first ones until the size of the new training set is complete.

Finally, the frequency of application of the generation-based strategy is determined based on the predictive power of the Pareto front, measured by the metric  $d_j$  described in the next section [85]. In other words, the update process will be delayed based on an update factor  $UF$  if each surrogate model achieves prediction values greater than 90% ( $\epsilon$ ). In contrast, if a surrogate model achieves prediction values below 90%, the update process will be executed earlier based on the obtained prediction. This parameter was

selected to guarantee a high fidelity in the subrogated model relative to the original model. Additionally, several authors recommend a value higher than 80% to guarantee competitive results [86,87].

---

### Algorithm 1 Surrogate model update procedure

---

**Require:**

$UG$ : generation number to apply the generation-based strategy  
 $UF$ : update factor  
 $\mathcal{P}_{train}^i$ : training set of the model  $i$   
 $\mathfrak{F}$ : Pareto front  
 $\varepsilon$ : prediction error threshold  
 $g$ : generation actual

- 1:  $\hat{Y} = surrogate\_evaluation(\mathfrak{F})$  ▷ Evaluation performed using surrogate model (Equation (5))
- 2:  $Y = original\_evaluation(\mathfrak{F})$  ▷ Evaluation performed using Equations (2)–(4)
- 3: **if**  $g == UG$  **then**
- 4:      $error = d_j(Y, \hat{Y})$  ▷  $d_j$  is computed using Equation (6)
- 5:     **if**  $error \leq \varepsilon$  **then**
- 6:          $\mathcal{P}_{train}^i = \mathcal{P}_{train}^i \cup \mathfrak{F}$
- 7:          $model\_train(i, \mathcal{P}_{train}^i)$  ▷ Train the model  $i$  using  $\mathcal{P}_{train}^i$
- 8:          $inc = round(UF * error, 0)$
- 9:          $UG = UG + inc$
- 10:     **else**
- 11:          $UF = UF * 2$

---

#### 2.5. Performance Metric for Surrogate Model Prediction

Evaluating surrogate models' predictive power is essential for implementing them, as this indicates their fidelity to the original models.

A metric used to evaluate the goodness of fit in regression models is the modified agreement index ( $d_j$ ) or the Willmott index [85]. The  $d_j$  values range from 0 to 1, where 0 indicates total disagreement between the observed and predicted values, and 1 indicates total agreement. Unlike  $R^2$ , which can generate negative values when comparing unrelated sets of values,  $d_j$  always produces results within the defined range. The  $d_j$  metric is calculated using Equation (6), where the observed and predicted values are denoted by  $Y_i$  and  $\hat{Y}_i$ , respectively, and the average of the observed values is represented by  $\bar{Y}$ . The value of  $j$  is usually set to one.

$$d_j = 1 - \frac{\sum_{i=1}^{|Y|} |Y_i - \hat{Y}_i|^j}{\sum_{i=1}^{|Y|} |\hat{Y}_i - \bar{Y}| + |Y_i - \bar{Y}|^j} \quad (6)$$

#### 2.6. Datasets

This study used 41 databases from the University of California Repository (UCR) to test our proposal [88]. Each database was donated by the original authors and categorized according to the problem addressed in their respective articles. Table 1 shows the databases utilized in this document.

Each database is classified into different domains based on the type of data input. For instance, data sets classified as image types comprise time series obtained from image outlines or other methods used to convert an image into a time series. On the contrary, the Spectro type includes a time series extracted from spectrograms of a specific food. Additionally, the sensor type comprises time series collected from various sensors, including engine noise, motion tracking, and electrical power metrics. The UCR repository [88] provides explanations and access to all types of datasets.

**Table 1.** Datasets used in this research. This data were obtained from [88]. The ‘Abbrev’ column is the authors’ suggested abbreviation for the database name.

Dataset	Abbrev.	Train Size	Test Size	Length	Classes Number	Type
ArrowHead	ARROW	36	175	251	3	IMAGE
CBF	CBF	30	900	128	3	SIMULATED
Coffee	COFFEE	28	28	286	2	SPECTRO
CricketX	CRICKX	390	390	300	12	MOTION
CricketY	CRICKY	390	390	300	12	MOTION
CricketZ	CRICKZ	390	390	300	12	MOTION
DiatomSizeReduction	DSR	16	306	345	4	IMAGE
DistalPhalanxOutlineAgeGroup	DPOAG	400	139	80	3	IMAGE
DistalPhalanxOutlineCorrect	DPOC	600	276	80	2	IMAGE
DistalPhalanxTW	DPTW	400	139	80	6	IMAGE
ECG200	ECG200	100	100	96	2	ECG
ECG5000	ECG5K	500	4500	140	5	ECG
ECGFiveDays	ECG5D	23	861	136	2	ECG
FaceAll	FALL	560	1690	131	14	IMAGE
FacesUCR	FUCR	200	2050	131	14	IMAGE
GunPoint	GUNP	50	150	150	2	MOTION
InsectWingbeatSound	IWS	220	1980	256	11	EPG
ItalyPowerDemand	ITAPD	67	1029	24	2	EPG
Lightning7	LIGHT7	70	73	319	7	SENSOR
MedicalImages	MEDIMG	381	760	99	10	HAR
MiddlePhalanxOutlineAgeGroup	MPOAG	400	154	80	3	SENSOR
MiddlePhalanxOutlineCorrect	MPOC	600	291	80	2	SENSOR
MiddlePhalanxTW	MPTW	399	154	80	6	OTHER
MoteStrain	MOTEST	20	1252	84	2	SIMULATED
PhalangesOutlinesCorrect	PHOC	1800	858	80	2	IMAGE
Plane	PLANE	105	105	144	7	IMAGE
ProximalPhalanxOutlineAgeGroup	PPOAG	400	205	80	3	AUDIO
ProximalPhalanxOutlineCorrect	PPOC	600	291	80	2	IMAGE
ProximalPhalanxTW	PPTW	400	205	80	6	SENSOR
SonyAIBORobotSurface1	SONY1	20	601	70	2	IMAGE
SonyAIBORobotSurface2	SONY2	27	953	65	2	MISC
SwedishLeaf	SWEDLF	500	625	128	15	SOUND
SyntheticControl	SYNCTR	300	300	60	6	SENSOR
ToeSegmentation1	TOES1	40	228	277	2	HEMODYNAMICS
ToeSegmentation2	TOES2	36	130	343	2	HEMODYNAMICS
Trace	TRACE	100	100	275	4	HEMODYNAMICS
TwoLeadECG	TWOECG	23	1139	82	2	DEVICE
TwoPatterns	TWOPAT	1000	4000	128	4	SENSOR
Wafer	WAFER	1000	6164	152	2	HAR
Wine	WINE	57	54	234	2	DEVICE
WordSynonyms	WORDS	267	638	270	25	AUDIO

### 3. Results and Discussion

This section outlines the experimental design used to assess our proposal’s efficacy across all dimensions. It also presents the findings of the experiments and their respective discussions. It is essential to mention that the methods compared in this section were implemented within the same experimental environment.

### 3.1. Experimental Design

The experiments performed and shown in the following sections attempt to analyze the behavior of our proposal comprehensively, covering (1) the predictive capability measuring the fidelity of the surrogated model concerning the original model, (2) the classification capability regarding the original approach, and (3) the classification capability concerning other widely used time series discretizers. Consequently, the experiments were planned to answer the following research questions based on the objectives defined in Section 1:

- Can sMODiTS increase the model fidelity regarding [42]? This question arises when analyzing the prediction power of sMODiTS and the proposal introduced in [42] compared to eMODiTS (original model). The results will seek to achieve the first research objective and are presented in Section 3.4.
- Is it possible to minimize the computational cost incurred by evaluating the solutions in the eMODiTS functions without losing the ability to classify the time series? This question arises to achieve the second research objective, which seeks to find an alternative evaluation of the objective functions without losing the time series classification rate. The answer to this question will be presented in Section 3.5.
- Is sMODiTS a competitive alternative compared to SAX-based symbolic discretization models? Finally, this question arises to analyze whether implementing the surrogate models in sMODiTS remains competitive in the task for which the tool was designed. Therefore, a comparison is made against symbolic discretization models showing competitive performance in time-series classification. In Section 3.6, the results that answer this question will be presented.

### 3.2. Parameter Settings

The parameters used for sMODiTS are the same as those recommended by the authors of eMODiTS [20], which were obtained through various tests with different configurations. These values are presented in Table 2. The parameter values of sMODiTS were taken from [42] for a fair comparison. These parameter values are described as follows. The model update process was performed at the beginning of every 60 generations for the generation-based strategy, starting with five updates and changing according to prediction ability. The prediction measure was  $d_j$ , and the prediction error threshold used to update the generation-based application was 0.9. Regarding the DTW parameter, the window size employed for the Sakoe-Chiba constraint was 10%; an often-used value when this method is implemented [89].

**Table 2.** Parameter settings for eMODiTS and sMODiTS. The values were selected according to those reported in [20].

Parameter	Value
Population size	100
Generation number	300
Number of independent executions	15
Crossover rate	80%
Mutation rate	20%

The EP approach was compared using the same sMODiTS experimental environment as that used for the SAX-based methods. However, since the other SAX-based approaches do not report results for all databases in the UCR repository, most databases have no parameter settings. Therefore, the word size and alphabet values for each method are obtained from the best solution found by EP. Furthermore, these symbolic discretization approaches require additional parameters, such as 1D-SAX and rSAX. For example, for 1D-SAX, the slope number used for its execution was eight, and for rSAX, the value of the parameter  $\tau$  was 10. For the other methods and their parameters, the initial settings

suggested by the authors were used due to the competitive performance observed in their specific applications.

### 3.3. Performance Measures

All methods were compared using the F-measure ( $FM$ ). This measure is one of the most robust and is insensitive to class balancing. The F-measure is expressed in Equation (7), where  $TP$  represents the number of true positives,  $FN$  represents the number of false negatives, and  $FP$  represents the number of false positives. Values close to one are preferred because they represent a suitable method performance.

$$FM = \frac{2TP}{2TP + FN + FP} \quad (7)$$

Moreover, the F-measure ratio ( $Fmeasure\_ratio$ ) between eMODiTS and sMODiTS, expressed in Equation (8), was used to interpret the data. When this ratio exceeds one, sMODiTS will achieve a higher F-measure value than eMODiTS and perform better. Otherwise, when this ratio is less than one, eMODiTS will have outperformed sMODiTS according to the F-measure. Finally, when the ratio is equal to one, it means that both methods will have the same performance in terms of classification.

$$Fmeasure\_ratio = \frac{sMODiTS\_Fmeasure}{eMODiTS\_Fmeasure} \quad (8)$$

Concerning statistical analysis, a normality test was applied for each data group in each comparison. It is essential to mention that the data used for this test were the misclassification rate reached using each method for all datasets. The normality test used was the *Anderson–Darling test*, with a confidence level of 95%. The data groups do not present a normal distribution for the first and last comparisons. Thus, the selected statistical test was the *Friedman test* with the *Nemenyi* post hoc test at a 95% confidence interval for both.

This confidence interval was chosen empirically, because a level of 95% is considered to be a balance between precision and reliability, meaning that the experiment, when repeated 95 times, will match the result. The opposite is true for the 90% and 99% intervals, where the reliability decreases for the former but increases for the latter, and the precision increases for the former but decreases for the latter [90].

One essential way to compare two Multi-Objective Evolutionary Algorithms (MOEA) is to compare the similarity among their final Pareto fronts. Consequently, a set of well-known metrics is employed to compare the Pareto fronts of eMODiTS and sMODiTS. These metrics are described below.

- **HVR [43]:** This metric is based on the hypervolume ( $H$ ) measure, which computes the volume in the space of objective functions covered by a set of non-dominated solutions based on a reference point. Therefore, the  $HVR$  computation is expressed in Equation (9), where  $H_{\mathfrak{F}_i}$  is the hypervolume of the obtained Pareto front and  $H_{\mathfrak{F}_j}$  is the hypervolume of the true Pareto front. In this document, we take the true Pareto front as the eMODiTS Pareto front and the obtained Pareto front as the sMODiTS Pareto front.  $HVR < 1$  indicates that the sMODiTS Pareto front does not reach the eMODiTS Pareto front,  $HVR = 1$  indicates that both fronts are similar, and  $HVR > 1$  indicates that the sMODiTS Pareto front outperforms the eMODiTS Pareto front. Therefore, the ideal value is  $HVR = 1$ .

$$HVR = \frac{H_{\mathfrak{F}_i}}{H_{\mathfrak{F}_j}} \quad (9)$$

- **Generational distance (GD) [91]:**  $GD$  measures the closeness of the obtained and true Pareto front. It is expressed in Equation (10), where  $|\mathfrak{F}_i|$  is the number of non-dominated solutions in the obtained Pareto front, and  $d_i$  is the Euclidean distance between each solution of the obtained Pareto front and the nearest solution of the True Pareto front, measured in the space of the objective functions. Like HVR, for our purpose, the true Pareto front is taken as the eMODiTS Pareto front, and the obtained

Pareto front is taken as the sMODiTS Pareto front. Values near zero indicate that the sMODiTS Pareto front is similar to the eMODiTS Pareto front. It is important to mention that the main reason for using this metric is to measure the closeness between the Pareto fronts. Therefore, it is not necessary to apply a data normalization technique to avoid bias in the distances of the points.

$$GD = \frac{\sqrt{\sum_{i=1}^{|\mathfrak{F}_i|} d_i^2}}{|\mathfrak{F}_i|} \quad (10)$$

- **Coverage measure (C)** [92]: This measure computes the portion of two Pareto fronts covered or dominated by one another or vice versa. Equation (11) is used to compute this measure, where  $C(\mathfrak{F}_i, \mathfrak{F}_j) = 1$  indicates that all elements of  $\mathfrak{F}_j$  are dominated by  $\mathfrak{F}_i$ , and  $C(\mathfrak{F}_i, \mathfrak{F}_j) = 0$  indicates that no elements from  $\mathfrak{F}_j$  are dominated by  $\mathfrak{F}_i$ . It is important to mention that  $C(\mathfrak{F}_i, \mathfrak{F}_j) \neq C(\mathfrak{F}_j, \mathfrak{F}_i)$  and  $C(\mathfrak{F}_i, \mathfrak{F}_j) \neq (1 - C(\mathfrak{F}_j, \mathfrak{F}_i))$ . Therefore, both scenarios must be analyzed to provide a wide view of this measure. Two Pareto fronts are considered as similar when the coverage in both senses is zero simultaneously.

$$C(\mathfrak{F}_i, \mathfrak{F}_j) = \frac{|\{\vec{y} \in \mathfrak{F}_j | \exists \vec{x} \in \mathfrak{F}_i : \vec{x} < \vec{y}\}|}{|\mathfrak{F}_j|} \quad (11)$$

- **Convergence index (CI)** [92]: This measure is the difference between the coverage of  $\mathfrak{F}_i$  over  $\mathfrak{F}_j$ . It is expressed by Equation (12), where  $CI(\mathfrak{F}_i, \mathfrak{F}_j) \geq 0$  denotes that the portion coverage of  $\mathfrak{F}_i$  is higher than  $\mathfrak{F}_j$ . Otherwise, the portion of  $\mathfrak{F}_j$  is higher than  $\mathfrak{F}_i$ .

$$CI(\mathfrak{F}_i, \mathfrak{F}_j) = C(\mathfrak{F}_i, \mathfrak{F}_j) - C(\mathfrak{F}_j, \mathfrak{F}_i) \quad (12)$$

Finally, for computational cost analysis, the percentage reduction in the number of evaluations performed was calculated using Equation (13), where  $sMODiTS\_Evals$  is the number of evaluations performed by sMODiTS and  $eMODiTS\_Evals$  is the number of evaluations performed by eMODiTS.

$$EvaluationsRatio = \frac{|sMODiTS\_Evals - eMODiTS\_Evals|}{eMODiTS\_Evals} * 100 \quad (13)$$

### 3.4. sMODiTS' Prediction Power Analysis

The sMODiTS' prediction power was measured based on the accumulated Pareto front obtained in each data set using the measures of root mean square error (RMSE),  $d_j$ , mean absolute percentage error (MAPE), and mean absolute error (MAE). It is essential to mention that each Pareto front point is a solution estimated from the surrogate model. Therefore, each point was evaluated in the original functions to perform the approximation analysis. Moreover, sMODiTS was compared with the previous work presented by Márquez-Grajales et al. [42] due to the intention of this work to improve its prediction power. Both were executed using five configurations, 1NNDTW, 3NNDTW, 5NNDTW, 7NNDTW, and 9NNDTW for  $k = \{1, 3, 5, 7, 9\}$  of the algorithm  $KNN_R$ , respectively, to find the best value for the parameter  $k$ . These values of  $k$  have been selected because several papers report better performance when  $k$  is less than or equal to nine [93–95]. The authors mention that with values higher than nine for the parameter  $k$ , the improvement in the results is minimal or null, and, on the contrary, with values between one and nine, the results show a significant performance in the problems where the KNN algorithm has been tested. Figure 9 and Table 3 show this analysis.

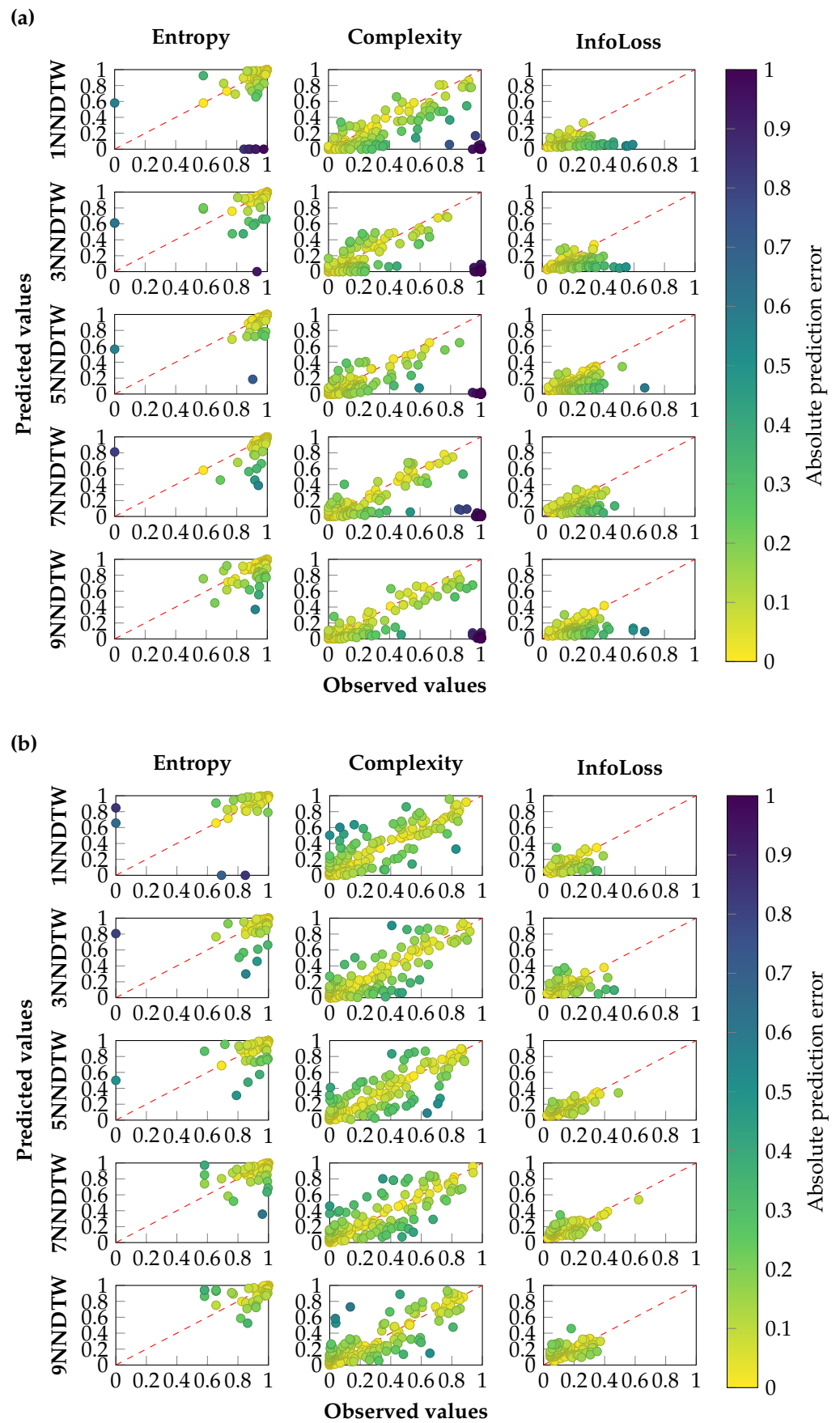


Figure 9. Prediction power reached by (a) Márquez-Grajales et al. [42] and (b) sMODiTS. The color bars represent the absolute discrepancy between the two approaches regarding the prediction error.



**Table 3.** Average prediction metrics achieved by Márquez-Grajales et al. [42] and sMODiTS. The numbers in parentheses represent the standard deviation of each measure and approach. Bold numbers represent the best values for each measure.

Function	Measure	Márquez-Grajales et al. [42]					sMODiTS				
		1NNDTW	3NNDTW	5NNDTW	7NNDTW	9NNDTW	1NNDTW	3NNDTW	5NNDTW	7NNDTW	9NNDTW
Entropy	RMSE	0.1243 (±0.1367)	0.0753 (±0.0624)	<b>0.0748</b> (±0.0693)	0.085 (±0.0582)	0.0843 (±0.0559)	0.1115 (±0.1107)	0.1099 (±0.1046)	0.0962 (±0.0745)	0.0984 (±0.0771)	0.0907 (±0.0811)
	$d_j$	0.491 (±0.1535)	0.405 (±0.1804)	0.3867 (±0.1674)	0.4557 (±0.1884)	0.4631 (±0.1687)	<b>0.6198</b> (±0.1344)	0.5916 (±0.1416)	0.5863 (±0.1388)	0.5882 (±0.114)	0.5719 (±0.1271)
	MAE	0.0594 (±0.0896)	<b>0.0318</b> (±0.043)	0.0343 (±0.0405)	0.0393 (±0.0386)	0.0402 (±0.0389)	0.0516 (±0.0736)	0.0573 (±0.0735)	0.045 (±0.0443)	0.0468 (±0.0439)	0.0437 (±0.0491)
Complexity	RMSE	0.2872 (±0.1245)	0.3352 (±0.1387)	0.3395 (±0.1243)	0.3481 (±0.1454)	0.3203 (±0.1319)	0.1371 (±0.0522)	0.1216 (±0.0368)	<b>0.1175</b> (±0.0366)	0.1248 (±0.0452)	0.125 (±0.0402)
	$d_j$	0.5719 (±0.1262)	0.5773 (±0.1059)	0.587 (±0.0831)	0.5998 (±0.1295)	0.6289 (±0.133)	0.6623 (±0.1846)	<b>0.6704</b> (±0.2032)	<b>0.6704</b> (±0.1712)	0.6629 (±0.1723)	0.6443 (±0.172)
	MAE	0.1721 (±0.0826)	0.1871 (±0.104)	0.1837 (±0.0839)	0.1983 (±0.11)	0.1742 (±0.0916)	0.0923 (±0.0359)	0.0835 (±0.0254)	<b>0.083</b> (±0.0261)	0.0883 (±0.0292)	0.0903 (±0.0288)
InfoLoss	RMSE	0.1185 (±0.0414)	0.1059 (±0.0419)	0.1017 (±0.035)	0.0954 (±0.0348)	0.1064 (±0.0459)	0.0628 (±0.0249)	0.0616 (±0.0276)	<b>0.0585</b> (±0.0173)	0.0631 (±0.0191)	0.0643 (±0.0176)
	$d_j$	0.4489 (±0.0703)	0.4955 (±0.093)	0.489 (±0.1232)	0.4798 (±0.1034)	0.4846 (±0.0677)	<b>0.5356</b> (±0.1152)	0.4663 (±0.1577)	0.44 (±0.1407)	0.4045 (±0.1444)	0.3769 (±0.1174)
	MAE	0.0868 (±0.0293)	0.0772 (±0.0324)	0.0733 (±0.0291)	0.068 (±0.0252)	0.0762 (±0.0317)	<b>0.0409</b> (±0.015)	0.0449 (±0.0211)	0.0433 (±0.0113)	0.0482 (±0.0138)	0.0493 (±0.0139)

Figure 9 shows the prediction plots in which the predicted values are compared with the original values. The color bar represents the absolute prediction error between the predicted and observed values, with a difference close to zero representing a high-accuracy prediction and a difference close to one representing a low-accuracy prediction. Moreover, a red dotted line is displayed to locate the scenario where the predicted and observed values match. Consequently, points close to or inside the red line represent a good approximation.

As we can see, in this figure, sMODiTS presents a higher accuracy in the approximation of the values of the objective functions, showing differences of less than 0.6 in the function of Complexity and Information Loss (InfoLoss), being entropy, the function with a higher difference value. In contrast, the approach presented by Márquez-Grajales et al. [42] presents a more significant number of differences with values greater than 0.8 in predicting both entropy and complexity, with information loss producing the most favorable outcomes. However, these values do not exceed those found by sMODiTS, whose plots are mainly around the red line in this function.

Moreover, these results are confirmed in Table 3, where the highest results are found for sMODiTS in most prediction measures. For example, the values of  $d_j$  using sMODiTS are highest in the 1NNDTW version for the entropy function (0.6198) and the information loss function (0.5356), while the 3NNDTW and 5NNDTW versions obtained the best value for the complexity function (0.6704). Regarding RMSE and MAE, only for the entropy function, the method introduced by Márquez-Grajales et al. [42] obtained lower values than those shown by sMODiTS (0.0748 and 0.0318 for RMSE and MAE, respectively). In contrast, the opposite is the case for the other functions, where sMODiTS is superior.

It is essential to mention that these results are due to the updating process of the proposed surrogate model, which evaluates its performance more frequently during the optimization process, achieving a more frequent update of the model fidelity compared to the one proposed by Márquez-Grajales et al. [42]. On the other hand, increasing the update frequency may affect the computational cost savings.

Consequently, a singular subrogated model cannot accurately represent all three functions. This situation is evidenced by distinct versions of  $KNN_R$  yield disparate outcomes across all three objective functions. Therefore, it is plausible to employ an alternative machine learning approach to align with the specific requirements of each function. Moreover, the outcomes above have enabled us to achieve the initial objective proposed in Section 1 and to answer the first research question mentioned before. It allows us to conclude that a more accurate approximation of the model can be achieved by modifying how the surrogate models are updated.

### 3.5. Comparison between eMODiTS and sMODiTS

The second analysis compares the performance between the original method (eMODiTS) and sMODiTS. Since both methods apply a multi-objective evolutionary algorithm, it is necessary to compare their performance based on the Pareto fronts obtained by both to validate that sMODiTS manages to approximate the front to the original one, which would mean that it has a similar performance but with a low computational cost. Therefore, this section also analyses the computational cost of the methods to verify that the Pareto fronts are similar and that the proposal has a lower cost.

Finally, it is necessary to validate the performance of sMODiTS concerning the time series classification task. For this purpose, a solution for both fronts was selected based on the technique that performed best for eMODiTS [20].

These experiments will provide a comprehensive understanding of how the surrogate model performs compared to the original model based on its effectiveness in addressing the study problem.

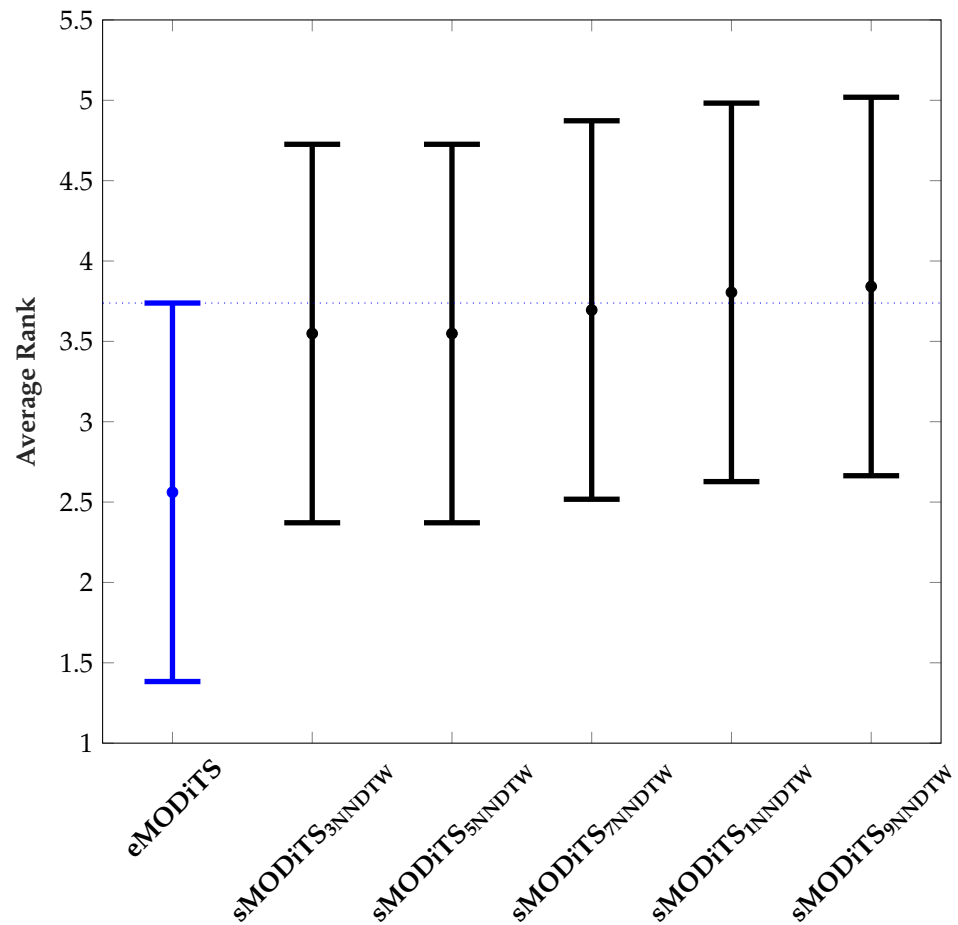
#### 3.5.1. Classification Performance

Figure 10 presents the results of the Friedman test combined with the Nemenyi post hoc test at a 95% confidence level to compare the original method (eMODiTS) against each surrogate model (sMODiTS). These tests were chosen because the data do not follow a normal distribution. As we can see, none of the compared methods had significant statistical differences; that is, all had the same behavior in terms of classification. These outcomes were anticipated, since the surrogate model approximates the actual model rather than enhancing it. Additionally, eMODiTS ranked better concerning the F-measure, which means that in most databases, eMODiTS still reaches better classification rates than sMODiTS. This behavior may be due to the initial sampling employed by sMODiTS, which does not guarantee that the initial training set covers the entire search space. Nevertheless, the search space in eMODiTS is vast and complex to cover using the conventional sampling methods due to the discretization scheme used by eMODiTS.

In particular, Table 4 shows the detailed results using the Wilcoxon Rank Sum Test statistic with a confidence level of 95% and the F-measure ratio (Equation (8)). This table shows that eMODiTS outperforms all sMODiTS versions in about half of the tested databases. However, in the remaining databases, sMODiTS manages to match the results with eMODiTS and even to outperform eMODiTS slightly in some of them due to the nature of the multi-objective evolutionary algorithm employed, which can find different solutions each time it is run because of its deterministic essence. On the other hand, when observing the range of values of the F-measure ratio, we can see that they are between 0.7 and 1.3, meaning that both methods are slightly close to each other, demonstrating that the surrogate models achieve an adequate approximation concerning the original model.

**Table 4.** F-measure ratio between eMODiTS and sMODiTS using the Wilcoxon rank sum test with a 95% confidence interval. Values in parentheses describe the *p*-value obtained from the statistical test, and the symbols in brackets indicate the type of interpretation of the results. The – symbol expresses that eMODiTS outperforms the sMODiTS version, and there is a statistically significant difference between them. The + symbol indicates that sMODiTS outperforms eMODiTS, and there is a significant statistical difference. On the other hand, the symbol \* indicates that eMODiTS outperforms sMODiTS, and there is no significant statistical difference. Finally, the symbol • indicates that sMODiTS outperforms eMODiTS, but there is no significant difference between them.

Dataset Abbrev.	sMODiTS <sub>1</sub> NNDTW	sMODiTS <sub>3</sub> NNDTW	sMODiTS <sub>5</sub> NNDTW	sMODiTS <sub>7</sub> NNDTW	sMODiTS <sub>9</sub> NNDTW
ARROW	0.7673 (0.002) [–]	0.8937 (0.0059) [–]	1.0192 (0.002) [+]	0.8457 (0.0039) [–]	0.9714 (0.625) [*]
CBF	1.1505 (0.002) [+]	0.9459 (0.002) [–]	1.041 (0.0098) [+]	0.9835 (0.002) [–]	0.9302 (0.002) [–]
COFFEE	0.8001 (0.002) [–]	0.8108 (0.002) [–]	0.8674 (0.0098) [–]	0.7831 (0.002) [–]	0.7819 (0.002) [–]
CRICKX	1.0237 (0.002) [+]	1.0436 (0.002) [+]	1.0371 (0.0195) [+]	1.026 (0.1309) [•]	1.0271 (0.3223) [•]
CRICKY	0.945 (0.002) [–]	0.9709 (0.002) [–]	1.0029 (0.084) [•]	0.986 (0.002) [–]	0.9803 (0.002) [–]
CRICKZ	0.9648 (0.002) [–]	0.9865 (0.2754) [*]	0.9889 (0.084) [*]	0.9541 (0.002) [–]	0.9867 (0.4922) [*]
DSR	0.9845 (0.1055) [*]	1.004 (0.4922) [•]	0.8962 (0.002) [–]	0.9512 (0.002) [–]	0.9578 (0.002) [–]
DPOAG	1.0013 (0.0645) [•]	1.003 (0.002) [+]	1.0023 (0.1055) [•]	1.0106 (0.002) [+]	0.9996 (0.0195) [–]
DPOC	0.938 (0.0039) [–]	0.9331 (0.002) [–]	0.8643 (0.0039) [–]	0.9904 (1) [*]	0.974 (0.9219) [*]
DPTW	0.9248 (0.002) [–]	0.938 (0.002) [–]	0.9285 (0.002) [–]	0.8387 (0.002) [–]	0.943 (0.002) [–]
ECG200	1.0613 (0.002) [+]	0.7807 (0.002) [–]	1.051 (0.0059) [+]	0.9615 (0.0371) [–]	0.9906 (0.2754) [*]
ECG5K	0.9483 (0.002) [–]	0.781 (0.002) [–]	0.9884 (0.2754) [*]	0.9598 (0.0039) [–]	0.9528 (0.3223) [*]
ECG5D	0.8682 (0.002) [–]	0.8093 (0.002) [–]	0.8213 (0.002) [–]	0.9557 (0.002) [–]	0.8446 (0.002) [–]
FALL	1.0291 (0.002) [+]	1.0321 (0.002) [+]	1.0194 (0.002) [+]	1.0194 (0.002) [+]	1.0272 (0.002) [+]
FUCR	0.8872 (0.002) [–]	0.9729 (0.002) [–]	0.9632 (0.002) [–]	0.8411 (0.002) [–]	0.9254 (0.002) [–]
GUNP	0.9483 (0.0039) [–]	0.8051 (0.002) [–]	0.9866 (0.002) [–]	0.8377 (0.002) [–]	0.8554 (0.002) [–]
IWS	1.0128 (0.002) [+]	1.021 (0.1309) [•]	0.9784 (0.0488) [–]	0.9906 (0.3223) [*]	1.0023 (0.0098) [+]
ITAPD	0.9986 (0.002) [–]	0.9677 (0.002) [–]	0.9909 (0.002) [–]	0.9812 (0.002) [–]	0.9578 (0.002) [–]
LIGHT7	1.0344 (0.002) [+]	1.0593 (0.002) [+]	0.9616 (0.002) [–]	1.0631 (0.0059) [+]	1.0093 (0.5566) [•]
MEDIMG	0.9456 (0.002) [–]	0.9861 (0.0039) [–]	1.1046 (0.002) [+]	0.8115 (0.002) [–]	1.0361 (0.002) [+]
MPOAG	0.9879 (0.0488) [–]	0.9946 (0.7695) [*]	1.0202 (0.002) [+]	0.9992 (0.625) [*]	1.0005 (0.1055) [•]
MPOC	0.9519 (0.002) [–]	0.8737 (0.002) [–]	0.9134 (0.002) [–]	0.9811 (0.7695) [*]	0.9661 (0.3223) [*]
MPTW	0.9976 (0.0039) [–]	1.0079 (0.0645) [•]	0.9782 (0.002) [–]	0.9926 (0.002) [–]	0.9972 (0.002) [–]
MOTEST	0.9822 (0.0137) [–]	1.0021 (0.625) [•]	0.9796 (0.002) [–]	0.9781 (0.002) [–]	1.0212 (0.4316) [•]
PHOC	1.0133 (0.1602) [•]	0.9992 (0.6953) [*]	0.9735 (0.002) [–]	0.9924 (0.002) [–]	0.9972 (0.0039) [–]
PLANE	0.9901 (0.4922) [*]	1.0193 (0.002) [+]	1.0207 (0.002) [+]	1.0333 (0.0039) [+]	1.0033 (0.0098) [+]
PPOAG	1.0367 (0.002) [+]	0.8407 (0.002) [–]	0.9065 (0.002) [–]	0.9885 (0.1602) [*]	0.9761 (0.0195) [–]
PPOC	0.8959 (0.002) [–]	0.9604 (0.0059) [–]	0.8479 (0.002) [–]	0.9167 (0.002) [–]	0.8842 (0.002) [–]
PPTW	0.941 (0.1055) [*]	0.9493 (0.1055) [*]	0.9696 (0.0039) [–]	0.9824 (0.8457) [*]	0.9511 (0.0273) [–]
SONY1	1.0087 (1) [•]	0.9582 (0.4922) [*]	1.011 (0.0039) [+]	1.1265 (0.002) [+]	1.0496 (0.002) [+]
SONY2	0.8905 (0.002) [–]	0.8561 (0.002) [–]	1.0303 (0.0645) [•]	1.0042 (0.0039) [+]	0.8561 (0.002) [–]
SWEDLF	0.9802 (0.0645) [*]	0.9437 (0.0039) [–]	0.8413 (0.002) [–]	1.0039 (0.0195) [+]	0.9045 (0.0137) [–]
SYNCTR	0.8787 (0.002) [–]	0.9206 (0.002) [–]	1.1072 (0.002) [+]	0.9225 (0.0039) [–]	0.9764 (0.0098) [–]
TOES1	0.8923 (0.002) [–]	0.883 (0.002) [–]	0.7864 (0.002) [–]	0.8268 (0.002) [–]	0.8292 (0.002) [–]
TOES2	1.0006 (0.0098) [+]	1.0001 (0.4922) [•]	0.9987 (0.002) [–]	1.0007 (0.3223) [•]	0.994 (0.002) [–]
TRACE	1.0966 (0.002) [+]	1.248 (0.0039) [+]	1.2622 (0.002) [+]	1.1132 (0.002) [+]	1.2332 (0.002) [+]
TWOECG	0.9385 (0.0371) [–]	1.1467 (0.002) [+]	1.0288 (0.002) [+]	0.8172 (0.002) [–]	0.888 (0.084) [*]
TWOPAT	1.0349 (1) [•]	1.0692 (0.002) [+]	0.9111 (0.002) [–]	0.9268 (0.002) [–]	1.0543 (0.1055) [•]
WAFER	0.8496 (0.002) [–]	0.9199 (0.002) [–]	0.7623 (0.002) [–]	0.8426 (0.002) [–]	0.9631 (0.6953) [*]
WINE	0.8519 (0.002) [–]	0.9554 (0.002) [–]	1.1025 (0.002) [+]	1.0133 (0.084) [•]	0.9276 (0.002) [–]
WORDS	0.7998 (0.002) [–]	0.8271 (0.0039) [–]	0.7934 (0.0039) [–]	0.8832 (0.084) [*]	0.5836 (0.002) [–]
<b>Total (–/+/*/•)</b>	<b>(24/9/4/4)</b>	<b>(23/8/5/5)</b>	<b>(23/13/2/3)</b>	<b>(23/8/7/3)</b>	<b>(22/6/8/5)</b>



**Figure 10.** Statistical comparison results between eMODiTS and every version of sMODiTS. The Friedman test and Nemenyi post hoc test were employed to perform this analysis with a 95% confidence level.

### 3.5.2. Analysis of Pareto Fronts

Table 5 shows the average results of applying five performance measures for Pareto front analysis in all test datasets. In this table, the analyzed Pareto fronts were obtained from eMODiTS, and each version of sMODiTS was used to compare the ability of the proposed surrogate model to approximate the solutions to the original model. In addition, this analysis allowed us to verify the similarity of both approaches in the objective function space.

**Table 5.** Analysis results of the Pareto fronts obtained using eMODiTS and each version of sMODiTS using the performance measures of HVR, Generational Distance (GD), coverage of the eMODiTS over sMODiTS ( $\vec{C}$ ), coverage the sMODiTS over eMODiTS ( $\overleftarrow{C}$ ), and convergence index (CI). The values displayed represent the average of each measure for all test databases, and the numbers in parentheses represent the standard deviation. In addition, the values in bold indicate the maximum values for each metric, while the values in italics indicate the minimum values.

Version	HVR	GD	$\vec{C}$	$\overleftarrow{C}$	CI
1NNDTW	<b>0.3891</b> ( $\pm 0.2111$ )	<i>0.0447</i> ( $\pm 0.0287$ )	0.9405 ( $\pm 0.0753$ )	0.0031 ( $\pm 0.0065$ )	0.9374 ( $\pm 0.0802$ )
3NNDTW	0.3454 ( $\pm 0.2009$ )	0.0477 ( $\pm 0.0332$ )	<i>0.9244</i> ( $\pm 0.0945$ )	<b>0.0062</b> ( $\pm 0.0121$ )	<i>0.9182</i> ( $\pm 0.1045$ )
5NNDTW	0.3341 ( $\pm 0.1806$ )	0.0503 ( $\pm 0.0321$ )	0.9394 ( $\pm 0.0703$ )	0.003 ( $\pm 0.0056$ )	0.9364 ( $\pm 0.0736$ )
7NNDTW	0.3566 ( $\pm 0.1996$ )	0.0523 ( $\pm 0.0382$ )	<b>0.943</b> ( $\pm 0.0721$ )	0.0037 ( $\pm 0.0088$ )	<b>0.9394</b> ( $\pm 0.0775$ )
9NNDTW	<i>0.334</i> ( $\pm 0.1892$ )	<b>0.0536</b> ( $\pm 0.0393$ )	0.9415 ( $\pm 0.0672$ )	<i>0.0029</i> ( $\pm 0.0061$ )	0.9386 ( $\pm 0.0716$ )

The HVR measure indicates that the space covered by eMODiTS is more significant than that covered by each version of sMODiTS. Nevertheless, the HVR mean value for

each dataset ranges between 0.334 and 0.3891, indicating that the eMODiTS hypervolume is slightly higher than the sMODiTS in the average case. Given that our MOOP is a minimization problem, the HVR results indicate that each Pareto front of the sMODiTS versions is above the eMODiTS Pareto front. These results suggest that the surrogate model approximation was not faithful to the original model.

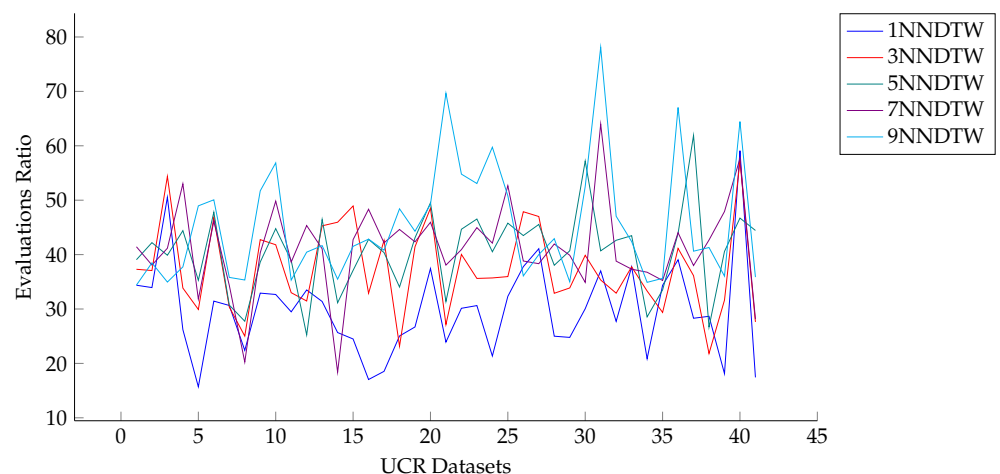
Regarding the coverage metric, the convergence index shows that the Pareto fronts generated by eMODiTS on the test datasets outperform those produced by all versions of sMODiTS, with values between 0.9182 and 0.9394. As with HVR, these values indicate that the eMODiTS Pareto fronts are situated below the sMODiTS fronts, which cover them in almost all databases tested. This result implies that sMODiTS failed to sufficiently approximate to obtain Pareto fronts similar to the original ones. However, it is essential to clarify that, due to its nature, sMODiTS could not improve the front obtained by eMODiTS. Its maximum performance would be to approach or equal the original fronts.

Despite the inability of sMODiTS, in all its versions, to obtain Pareto fronts comparable to the original, it does succeed in approximating them. This situation can be observed in the maximum (0.0447) and minimum (0.0536) values of the GD measure, where their average values are close to zero, indicating that the fronts are very close.

In conclusion, sMODiTS cannot generate Pareto fronts that are identical to eMODiTS. However, it can achieve a similar outcome, maintaining the same behavioral characteristics.

### 3.5.3. Computational Cost Analysis

The computational cost was measured based on the number of evaluations performed in eMODiTS and sMODiTS. Figure 11 shows the number of reduction percentages of evaluations based on Equation (13) reached by sMODiTS compared with eMODiTS. eMODiTS performs 30,200 calls per objective function in one execution, while sMODiTS computes an average of 18,389 calls per objective function, representing a reduction of 40%. It is essential to mention that the number of evaluations executed by sMODiTS varies due to the model updating process. As mentioned in Section 2.4.3, the training set update process in sMODiTS consists of evaluating the first Pareto front in original objective functions and inserting them into this set. Since the number of solutions for the first Pareto front differs, in each update, sMODiTS performs different calls of the original objective functions.



**Figure 11.** Percentage reduction in the number of evaluations reached by sMODiTS compared to eMODiTS.

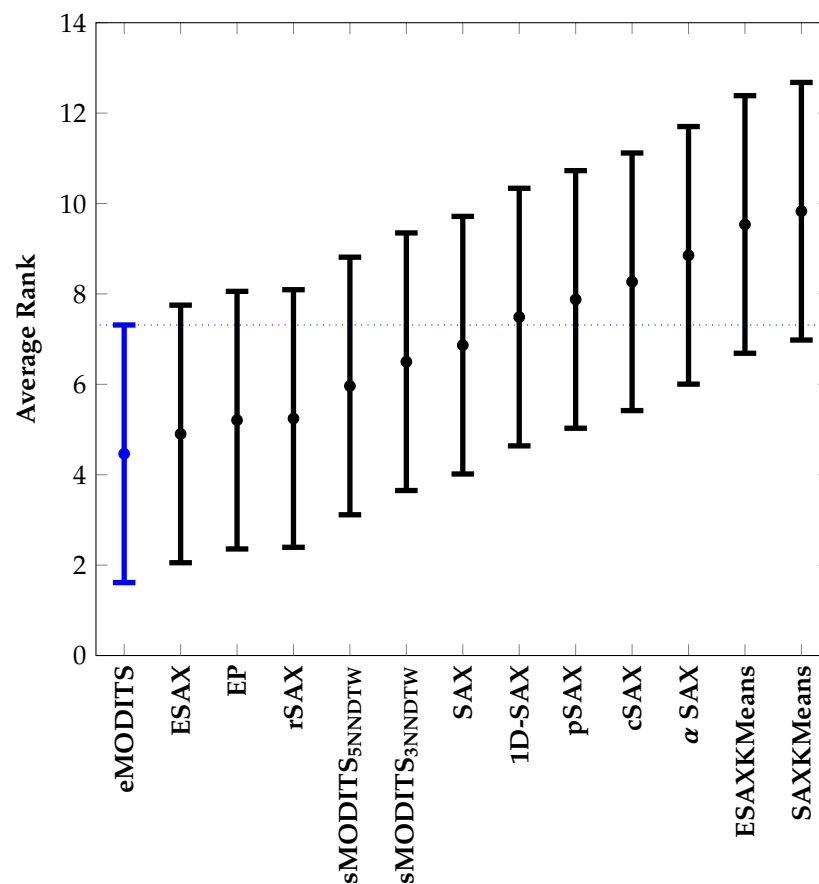
As we can see, the reduction percentages of evaluations performed by sMODiTS range between 15% and 80%. This reduction means that sMODiTS performs fewer of the original objective function calls, reducing the computational cost of these functions. However, since sMODiTS uses a surrogate model based on DTW distance, the computational cost is transferred to this process, increasing the algorithm's execution time.

The examination conducted in this section enables us to achieve the second goal outlined in Section 1 and address the second question mentioned in Section 3.1.

### 3.6. Comparison of sMODiTS among the SAX-Based Methods

The last experiment compares sMODiTS against SAX-based approaches to analyze if the surrogate model incorporated in eMODiTS is still conserving the classification performance when its discretization scheme is applied to a temporary dataset. Since all the data obtained for each method do not follow a normal distribution, the Friedman statistical test was used with the Nemenyi post hoc test with a confidence level of 95%. The SAX-based models used in this comparison were EP, SAX,  $\alpha$ SAX, ESAX, ESAXKMeans, 1D-SAX, SAXKMeans, rSAX, pSAX, and cSAX. These approaches were selected because of their competitive results and novel discretization schemes. In the case of sMODiTS, the 3NNDTW and 5NNDTW versions were identified as the most appropriate for inclusion in this experiment based on Figure 10 because they reported a close ranking between them and generated the most minor difference concerning eMODiTS.

Figure 12 shows the results of the statistical tests of the comparison of sMODiTS with SAX-based approaches. eMODiTS was used as a reference for this purpose. This figure shows no significant difference among all the methods, which means that sMODiTS presents a similar behavior in the classification task to the well-known symbolic discretization methods, preserving the characteristics of the original approach (eMODiTS).



**Figure 12.** Statistical comparison results between two versions of sMODiTS and ten SAX-based approaches. eMODiTS was used as a reference. The Friedman test and Nemenyi post hoc test were employed to perform this analysis with a 95% confidence level.

As previously mentioned, the outcomes detailed in these sections are anticipated, since sMODiTS solely employs surrogate models to estimate the objective functions for identifying appropriate symbolic discretization schemes without introducing any concep-

tual alterations to the original method. Using surrogate models has proven to be a practical approach for reducing the evaluation cost of the objective functions.

These results answer the third research question described in Section 3.1 and consequently the third goal of Section 1.

#### 4. Conclusions

Surrogate models are an alternative tool for approximating objective functions in evolutionary optimization. This document implemented surrogate models for estimating the objective functions of eMODiTS. This research is an extension of the approach proposed by Márquez-Grajales et al. [42]. Since eMODiTS employs individuals of different sizes, the KNN algorithm and DTW were incorporated as surrogate models to estimate the objective function values of eMODiTS. This surrogate-assisted version was called sMODiTS, and its behavior was compared against the original model.

The results suggest that each version of sMODiTS implemented behaves similarly to the original approach (eMODiTS), which does not present a significant statistical difference. However, our proposal presented a higher error classification than the original model. Moreover, regarding the prediction power of the surrogate model, the metrics suggest that sMODiTS presents a low accuracy in estimating the values of the original fitness functions.

On the other hand, the Pareto fronts of both approaches were compared using MOEA performance measures to evaluate the behavior of the final solutions found by each approach. These measures indicate that the performance of our approach algorithm is similar to the original algorithm since the two fronts are close to each other.

Regarding computational cost, the number of evaluations performed by the surrogate version is lower than those achieved by the original, with reduction percentages in the use of the original objective functions between 15% and 80%, decreasing the computational cost of the original algorithm.

Finally, the statistical test indicates that sMODiTS achieves competitive results in classification terms compared to SAX-based symbolic discretization methods because there is no statistical difference among all the compared methods, ranking less than seven approaches out of ten.

In summary, the surrogate models used in this study approximated the actual model outcomes while significantly reducing the number of evaluations of the objective functions, which are computationally expensive to calculate. They also preserved the effectiveness of the time series discretization task compared to methods that have demonstrated competitive performance in tested problems. Furthermore, although the surrogate models' accuracy is low, they are suitable for problems where the solutions have different lengths from each other, particularly in the time series discretization proposed by the eMODiTS approach. Consequently, the objectives set out in Section 1 have been achieved, verifying that the surrogate model maintains the original model results with a competitive approximation but with a lower computational cost.

As a future study, we propose to implement other surrogate models (one different per each objective function) that are capable of handling different-sized solutions to increase the sMODiTS' estimation accuracy concerning the fitness functions of the eMODiTS method. Moreover, suitable initial sampling methods can be incorporated into this solution to achieve a reliable approximation of the original model. Finally, a comparison of different training set codifications can be performed to evaluate if this feature impacts the fidelity of the original models.

**Author Contributions:** Conceptualization, A.M.-G., E.M.-M. and H.-G.A.-M.; formal analysis, A.M.-G. and F.S.-M.; methodology, A.M.-G., E.M.-M. and H.-G.A.-M.; software, A.M.-G.; writing—original draft, A.M.-G.; writing—review and editing, E.M.-M., H.-G.A.-M. and F.S.-M. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The original data presented in the study are openly available from the UCR repository at [88].

**Acknowledgments:** The first author is grateful to the Consejo Nacional de Humanidades, Ciencias y Tecnología (CONAHCYT) for the postdoctoral grant awarded for this research under CVU number 419862. The authors would like to thank Eamonn Keogh and all the people who collected, preprocessed, and donated the data for the UCR TSC archive.

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

## References

- Dimitrova, E.S.; Licona, M.P.V.; McGee, J.; Laubenbacher, R. Discretization of time series data. *J. Comput. Biol.* **2010**, *17*, 853–868. [[CrossRef](#)] [[PubMed](#)]
- Lin, J.; Keogh, E.; Lonardi, S.; Chiu, B. A symbolic representation of time series, with implications for streaming algorithms. In Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, San Diego, CA, USA, 13 June 2003; pp. 2–11.
- Lkhagva, B.; Suzuki, Y.; Kawagoe, K. Extended SAX: Extension of symbolic aggregate approximation for financial time series data representation. In Proceedings of the DEWS2006, Ginowan, Japan, 1–3 March 2006.
- Sant’Anna, A.; Wickström, N. Symbolization of time-series: An evaluation of sax, persist, and aca. In Proceedings of the 2011 4th International Congress on Image and Signal Processing, Shanghai, China, 15–17 October 2011; IEEE: Piscataway, NJ, USA, 2011; Volume 4, pp. 2223–2228.
- Zhang, H.; Dong, Y.; Xu, D. Entropy-based Symbolic Aggregate Approximation Representation Method for Time Series. In Proceedings of the 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 11–13 December 2020; Volume 9, pp. 905–909. [[CrossRef](#)]
- Muhammad Fuad, M.M. Modifying the Symbolic Aggregate Approximation Method to Capture Segment Trend Information. In *Modeling Decisions for Artificial Intelligence*; Torra, V., Narukawa, Y., Nin, J., Agell, N., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 230–239.
- Hui, R.; Xiaoguang, H.; Jin, X.; Guofeng, Z. TrSAX—An improved time series symbolic representation for classification. *ISA Trans.* **2020**, *100*, 387–395. [[CrossRef](#)]
- Lkhagva, B.; Suzuki, Y.; Kawagoe, K. New time series data representation ESAX for financial applications. In Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW’06), Atlanta, GA, USA, 3–7 April 2006; IEEE: Piscataway, NJ, USA, 2006; p. x115.
- Pham, N.D.; Le, Q.L.; Dang, T.K. Two novel adaptive symbolic representations for similarity search in time series databases. In Proceedings of the 2010 12th International Asia-Pacific Web Conference, Busan, Republic of Korea, 6–8 April 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 181–187.
- Bai, X.; Xiong, Y.; Zhu, Y.; Zhu, H. Time series representation: A random shifting perspective. In Proceedings of the International Conference on Web-Age Information Management, Beidaihe, China, 14–16 June 2013; Springer: Cham, Switzerland, 2013; pp. 37–50.
- Malinowski, S.; Guyet, T.; Quiniou, R.; Tavenard, R. 1d-sax: A novel symbolic representation for time series. In Proceedings of the International Symposium on Intelligent Data Analysis, London, UK, 17–19 October 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 273–284.
- He, Z.; Zhang, C.; Ma, X.; Liu, G. Hexadecimal Aggregate Approximation Representation and Classification of Time Series Data. *Algorithms* **2021**, *14*, 353. [[CrossRef](#)]
- Kegel, L.; Hartmann, C.; Thiele, M.; Lehner, W. Season- and Trend-aware Symbolic Approximation for Accurate and Efficient Time Series Matching. *Datenbank-Spektrum* **2021**, *21*, 225–236. [[CrossRef](#)]
- Bountrogiannis, K.; Tzagkarakis, G.; Tsakalides, P. Data-driven kernel-based probabilistic SAX for time series dimensionality reduction. In Proceedings of the 2020 28th European Signal Processing Conference (EUSIPCO), Amsterdam, The Netherlands, 18–21 January 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 2343–2347.
- dos Santos Passos, H.; Teodoro, F.G.S.; Duru, B.M.; de Oliveira, E.L.; Peres, S.M.; Lima, C.A. Symbolic representations of time series applied to biometric recognition based on ecg signals. In Proceedings of the 2017 international joint conference on neural networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 3199–3207.
- Acosta-Mesa, H.G.; Rechy-Ramírez, F.; Mezura-Montes, E.; Cruz-Ramírez, N.; Jiménez, R.H. Application of time series discretization using evolutionary programming for classification of precancerous cervical lesions. *J. Biomed. Inform.* **2014**, *49*, 73–83. [[CrossRef](#)]
- Ahmed, A.M.; Bakar, A.A.; Hamdan, A.R. Harmony search algorithm for optimal word size in symbolic time series representation. In Proceedings of the 2011 3rd Conference on Data Mining and Optimization (DMO), Putrajaya, Malaysia, 28–29 June 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 57–62.
- Fuad, M.; Marwan, M. Genetic algorithms-based symbolic aggregate approximation. In Proceedings of the International Conference on Data Warehousing and Knowledge Discovery, Vienna, Austria, 3–6 September 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 105–116.



19. Fuad, M.M.M. Differential evolution versus genetic algorithms: Towards symbolic aggregate approximation of non-normalized time series. In Proceedings of the 16th International Database Engineering & Applications Symposium, IDEAS '12, Prague, Czech Republic, 8–10 August 2012; Association for Computing Machinery: New York, NY, USA, 2012; pp. 205–210. [\[CrossRef\]](#)
20. Márquez-Grajales, A.; Acosta-Mesa, H.G.; Mezura-Montes, E.; Graff, M. A multi-breakpoints approach for symbolic discretization of time series. *Knowl. Inf. Syst.* **2020**, *62*, 2795–2834. [\[CrossRef\]](#)
21. Jiang, P.; Zhou, Q.; Shao, X. *Surrogate Model-Based Engineering Design and Optimization*; Springer: Singapore, 2020.
22. Koziel, S.; Pietrenko-Dabrowska, A. Rapid multi-criterial antenna optimization by means of pareto front triangulation and interpolative design predictors. *IEEE Access* **2021**, *9*, 35670–35680. [\[CrossRef\]](#)
23. Koziel, S.; Pietrenko-Dabrowska, A. Fast multi-objective optimization of antenna structures by means of data-driven surrogates and dimensionality reduction. *IEEE Access* **2020**, *8*, 183300–183311. [\[CrossRef\]](#)
24. Koziel, S.; Pietrenko-Dabrowska, A. Constrained multi-objective optimization of compact microwave circuits by design triangulation and pareto front interpolation. *Eur. J. Oper. Res.* **2022**, *299*, 302–312. [\[CrossRef\]](#)
25. Pietrenko-Dabrowska, A.; Koziel, S. Accelerated multiobjective design of miniaturized microwave components by means of nested kriging surrogates. *Int. J. RF Microw. Comput.-Aided Eng.* **2020**, *30*, e22124. [\[CrossRef\]](#)
26. do Nascimento, C.A.R.; Mariani, V.C.; dos Santos Coelho, L. Integrative numerical modeling and thermodynamic optimal design of counter-flow plate-fin heat exchanger applying neural networks. *Int. J. Heat Mass Transf.* **2020**, *159*, 120097. [\[CrossRef\]](#)
27. Amrit, A.; Leifsson, L.; Koziel, S. Fast multi-objective aerodynamic optimization using sequential domain patching and multifidelity models. *J. Aircr.* **2020**, *57*, 388–398. [\[CrossRef\]](#)
28. Zhu, H.; Li, D.; Nie, H.; Wei, X.; Wei, Y. Multiobjective optimization of a staggered-rotor octocopter design based on a surrogate model. *Aerosp. Sci. Technol.* **2023**, *139*, 108387. [\[CrossRef\]](#)
29. Fan, Y.; Lu, W.; Miao, T.; Li, J.; Lin, J. Multiobjective optimization of the groundwater exploitation layout in coastal areas based on multiple surrogate models. *Environ. Sci. Pollut. Res.* **2020**, *27*, 19561–19576. [\[CrossRef\]](#)
30. Ji, X.; Zhang, Y.; Gong, D.; Sun, X.; Guo, Y. Multisurrogate-Assisted Multitasking Particle Swarm Optimization for Expensive Multimodal Problems. *IEEE Trans. Cybern.* **2023**, *53*, 2516–2530. [\[CrossRef\]](#)
31. Ye, P.; Pan, G. An optimization algorithm combining local exploitation and global exploration for computationally expensive problems. *Soft Comput.* **2024**, *28*, 7841–7860. [\[CrossRef\]](#)
32. Zhao, F.; Tang, Z.; Xiao, Z.; Liu, H.; Fan, J.; Li, L. Ensemble CART surrogate-assisted automatic multi-objective rough fuzzy clustering algorithm for unsupervised image segmentation. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108104. [\[CrossRef\]](#)
33. Shui, Y.; Li, H.; Sun, J.; Zhang, Q. Approximating robust Pareto fronts by the MEOF-based multiobjective evolutionary algorithm with two-level surrogate models. *Inf. Sci.* **2024**, *657*, 119946. [\[CrossRef\]](#)
34. Sampaio, P.R. DEFT-FUNNEL: An open-source global optimization solver for constrained grey-box and black-box problems. *Comput. Appl. Math.* **2021**, *40*, 176. [\[CrossRef\]](#)
35. Romor, F.; Tezzele, M.; Rozza, G. A Local Approach to Parameter Space Reduction for Regression and Classification Tasks. *J. Sci. Comput.* **2024**, *99*, 83. [\[CrossRef\]](#)
36. Araújo, G.; Gomes, R.; Ferrão, P.; Gomes, M.G. Optimizing building retrofit through data analytics: A study of multi-objective optimization and surrogate models derived from energy performance certificates. *Energy Built Environ.* **2024**, *5*, 889–899. [\[CrossRef\]](#)
37. Greiner, D.; Cacereño, A. Enhancing the maintenance strategy and cost in systems with surrogate assisted multiobjective evolutionary algorithms. *Dev. Built Environ.* **2024**, *19*, 100478. [\[CrossRef\]](#)
38. Hook, D.W.; Porter, S.J.; Herzog, C. Dimensions: Building Context for Search and Evaluation. *Front. Res. Metrics Anal.* **2018**, *3*, 23. [\[CrossRef\]](#)
39. Dumont, V.; Garner, C.; Trivedi, A.; Jones, C.; Ganapati, V.; Mueller, J.; Perciano, T.; Kiran, M.; Day, M. HYPPO: A Surrogate-Based Multi-Level Parallelism Tool for Hyperparameter Optimization. In Proceedings of the 2021 IEEE/ACM Workshop on Machine Learning in High Performance Computing Environments (MLHPC), St. Louis, MO, USA, 15 November 2021; pp. 81–93. [\[CrossRef\]](#)
40. Wu, Z.Y.; Rahman, A. Optimized Deep Learning Framework for Water Distribution Data-Driven Modeling. *Procedia Eng.* **2017**, *186*, 261–268. [\[CrossRef\]](#)
41. Vijayaprabakaran, K.; Sathiyamurthy, K. Neuroevolution based hierarchical activation function for long short-term model network. *J. Ambient Intell. Humaniz. Comput.* **2021**, *12*, 10757–10768. [\[CrossRef\]](#)
42. Márquez-Grajales, A.; Mezura-Montes, E.; Acosta-Mesa, H.G.; Salas-Martínez, F. Use of a Surrogate Model for Symbolic Discretization of Temporal Data Sets through eMODiTS and a Training Set with Varying-Sized Instances. In *Advances in Computational Intelligence*; MICAI 2023 International Workshops; Calvo, H., Martínez-Villaseñor, L., Ponce, H., Zatarain Cabada, R., Montes Rivera, M., Mezura-Montes, E., Eds.; Springer Nature: Cham, Switzerland, 2024; pp. 360–372.
43. Coello, C.A.C.; Lamont, G.B.; Veldhuizen, D.A.V. *Evolutionary Algorithms for Solving Multi-Objective Problems*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 5.
44. Deb, K. *Multi-Objective Optimization using Evolutionary Algorithms*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2001.
45. Peng, H.; Pi, C.; Xiong, J.; Fan, D.; Shen, F. A dynamic multi-objective evolutionary algorithm with variable stepsize and dual prediction strategies. *Future Gener. Comput. Syst.* **2024**, *161*, 390–403. [\[CrossRef\]](#)

46. Rangaiah, G.P. *Multi-Objective Optimization: Techniques and Applications in Chemical Engineering*; World Scientific: Singapore, 2016; Volume 5.
47. Delboeuf, J. *Mathematical Psychics, an Essay on the Application of Mathematics to the Moral Sciences*; CK Paul: London, UK, 1881.
48. Deb, K.; Deb, K. Multi-objective Optimization. In *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*; Springer: Boston, MA, USA, 2014; Chapter 15, pp. 403–449. [[CrossRef](#)]
49. Pareto, V. *Cours d'économie Politique*; Librairie Droz: Geneva, Switzerland, 1964; Volume 1.
50. Deb, K.; Agrawal, S.; Pratap, A.; Meyarivan, T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Proceedings of the International Conference on Parallel Problem Solving from Nature, Paris, France, 18–20 September 2000*; Springer: Berlin/Heidelberg, Germany, 2000; pp. 849–858.
51. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
52. Verma, S.; Pant, M.; Snasel, V. A comprehensive review on NSGA-II for multi-objective combinatorial optimization problems. *IEEE Access* **2021**, *9*, 57757–57791. [[CrossRef](#)]
53. Syswerda, G. Uniform crossover in genetic algorithms. In *Proceedings of the Third International Conference on Genetic Algorithms, Fairfax, VA, USA, 4–7 June 1989*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1989; Volume 3, pp. 2–9.
54. Poli, R.; Langdon, W.B. Genetic programming with one-point crossover. In *Soft Computing in Engineering Design and Manufacturing*; Springer: Berlin/Heidelberg, Germany, 1998; pp. 180–189.
55. Mirjalili, S. Genetic algorithm. In *Evolutionary Algorithms and Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 43–55.
56. Zainuddin, F.A.; Abd Samad, M.F.; Tunggal, D. A review of crossover methods and problem representation of genetic algorithm in recent engineering applications. *Int. J. Adv. Sci. Technol.* **2020**, *29*, 759–769.
57. Singh, G.; Gupta, N. A Study of Crossover Operators in Genetic Algorithms. In *Frontiers in Nature-Inspired Industrial Optimization*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 17–32.
58. Singh, A.; Gupta, N.; Sinhal, A. Artificial bee colony algorithm with uniform mutation. In *Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011), Roorkee, India, 20–22 December 2011*; Springer: Roorkee, India, 2012; pp. 503–511.
59. Koziel, S.; Ciaurri, D.E.; Leifsson, L. Surrogate-based methods. In *Computational Optimization, Methods and Algorithms*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 33–59.
60. Tong, H.; Huang, C.; Minku, L.L.; Yao, X. Surrogate models in evolutionary single-objective optimization: A new taxonomy and experimental study. *Inf. Sci.* **2021**, *562*, 414–437. [[CrossRef](#)]
61. Miranda-Varela, M.E.; Mezura-Montes, E. Constraint-handling techniques in surrogate-assisted evolutionary optimization. An empirical study. *Appl. Soft Comput.* **2018**, *73*, 215–229. [[CrossRef](#)]
62. Bhosekar, A.; Ierapetritou, M. Advances in surrogate based modeling, feasibility analysis, and optimization: A review. *Comput. Chem. Eng.* **2018**, *108*, 250–267. [[CrossRef](#)]
63. Fang, K.; Liu, M.Q.; Qin, H.; Zhou, Y.D. *Theory and Application of Uniform Experimental Designs*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 221.
64. Kai-Tai, F.; Dennis K.J., L.; Peter, W.; Yong, Z. Uniform Design: Theory and Application. *Technometrics* **2000**, *42*, 237–248. [[CrossRef](#)]
65. Yondo, R.; Andrés, E.; Valero, E. A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses. *Prog. Aerosp. Sci.* **2018**, *96*, 23–61. [[CrossRef](#)]
66. Díaz-Manríquez, A.; Toscano, G.; Barron-Zambrano, J.H.; Tello-Leal, E. A review of surrogate assisted multiobjective evolutionary algorithms. *Comput. Intell. Neurosci.* **2016**, *2016*, 9420460. [[CrossRef](#)] [[PubMed](#)]
67. Deb, K.; Roy, P.C.; Hussein, R. Surrogate modeling approaches for multiobjective optimization: Methods, taxonomy, and results. *Math. Comput. Appl.* **2020**, *26*, 5. [[CrossRef](#)]
68. Lv, Z.; Wang, L.; Han, Z.; Zhao, J.; Wang, W. Surrogate-assisted particle swarm optimization algorithm with Pareto active learning for expensive multi-objective optimization. *IEEE/CAA J. Autom. Sin.* **2019**, *6*, 838–849. [[CrossRef](#)]
69. Zhao, M.; Zhang, K.; Chen, G.; Zhao, X.; Yao, C.; Sun, H.; Huang, Z.; Yao, J. A surrogate-assisted multi-objective evolutionary algorithm with dimension-reduction for production optimization. *J. Pet. Sci. Eng.* **2020**, *192*, 107192. [[CrossRef](#)]
70. Wang, X.; Jin, Y.; Schmitt, S.; Olhofer, M. An adaptive Bayesian approach to surrogate-assisted evolutionary multi-objective optimization. *Inf. Sci.* **2020**, *519*, 317–331. [[CrossRef](#)]
71. Ruan, X.; Li, K.; Derbel, B.; Liefoghe, A. Surrogate assisted evolutionary algorithm for medium scale multi-objective optimisation problems. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference, Cancún, Mexico, 8–12 July 2020*; pp. 560–568.
72. Bao, K.; Fang, W.; Ding, Y. Adaptive Weighted Strategy Based Integrated Surrogate Models for Multiobjective Evolutionary Algorithm. *Comput. Intell. Neurosci.* **2022**, *2022*, 5227975. [[CrossRef](#)]
73. Wang, X.; Jin, Y.; Schmitt, S.; Olhofer, M. Transfer Learning Based Co-Surrogate Assisted Evolutionary Bi-Objective Optimization for Objectives with Non-Uniform Evaluation Times. *Evol. Comput.* **2022**, *30*, 221–251. [[CrossRef](#)] [[PubMed](#)]

74. Rosales-Pérez, A.; Coello, C.A.C.; Gonzalez, J.A.; Reyes-Garcia, C.A.; Escalante, H.J. A hybrid surrogate-based approach for evolutionary multi-objective optimization. In Proceedings of the 2013 IEEE Congress on Evolutionary Computation, Cancun, Mexico, 20–23 June 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 2548–2555.
75. Bi, Y.; Xue, B.; Zhang, M. Instance Selection-Based Surrogate-Assisted Genetic Programming for Feature Learning in Image Classification. *IEEE Trans. Cybern.* **2021**, *53*, 1118–1132. [[CrossRef](#)] [[PubMed](#)]
76. Blank, J.; Deb, K. GPSAF: A Generalized Probabilistic Surrogate-Assisted Framework for Constrained Single-and Multi-objective Optimization. *arXiv* **2022**, arXiv:2204.04054.
77. Wu, M.; Wang, L.; Xu, J.; Wang, Z.; Hu, P.; Tang, H. Multiobjective ensemble surrogate-based optimization algorithm for groundwater optimization designs. *J. Hydrol.* **2022**, *612*, 128159. [[CrossRef](#)]
78. Isaacs, A.; Ray, T.; Smith, W. An evolutionary algorithm with spatially distributed surrogates for multiobjective optimization. In Proceedings of the Australian Conference on Artificial Life, Gold Coast, Australia, 4–6 December 2007; Springer: New York, NY, USA, 2007; pp. 257–268.
79. Datta, R.; Regis, R.G. A surrogate-assisted evolution strategy for constrained multi-objective optimization. *Expert Syst. Appl.* **2016**, *57*, 270–284. [[CrossRef](#)]
80. Kourakos, G.; Mantoglou, A. Development of a multi-objective optimization algorithm using surrogate models for coastal aquifer management. *J. Hydrol.* **2013**, *479*, 13–23. [[CrossRef](#)]
81. Yan, Y.; Zhang, Y.; Hu, W.; Guo, X.j.; Ma, C.; Wang, Z.a.; Zhang, Q. A multiobjective evolutionary optimization method based critical rainfall thresholds for debris flows initiation. *J. Mt. Sci.* **2020**, *17*, 1860–1873. [[CrossRef](#)]
82. De Melo, M.C.; Santos, P.B.; Faustino, E.; Bastos-Filho, C.J.; Sodr e, A.C. Computational Intelligence-Based Methodology for Antenna Development. *IEEE Access* **2021**, *10*, 1860–1870. [[CrossRef](#)]
83. Gatopoulos, I.; Lepert, R.; Wiggers, A.; Mariani, G.; Tomczak, J. Evolutionary Algorithm with Non-parametric Surrogate Model for Tensor Program optimization. In Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, 19–24 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.
84. Zhi, J.; Yong, Z.; Xian-fang, S.; Chunlin, H. A Surrogate-Assisted Ensemble Particle Swarm Optimizer for Feature Selection Problems. In *Advances in Swarm Intelligence, International Conference on Sensing and Imaging*; Springer: Cham, Switzerland, 2022; pp. 160–166.
85. Legates, D.R.; McCabe, G.J., Jr. Evaluating the use of “goodness-of-fit” Measures in hydrologic and hydroclimatic model validation. *Water Resour. Res.* **1999**, *35*, 233–241. [[CrossRef](#)]
86. Yang, Q.; Wang, H.; Yang, C.; Wang, Y.; Hu, D.; Wang, B.; Duan, B. Research on surrogate models and optimization algorithms of compressor characteristic based on digital twins. *J. Eng. Res.* **2024**. [[CrossRef](#)]
87. Jiang, P.; Zhou, Q.; Shao, X. Surrogate-Model-Based Design and Optimization. In *Surrogate Model-Based Engineering Design and Optimization*; Springer: Singapore, 2020; pp. 135–236. [[CrossRef](#)]
88. Dau, H.A.; Keogh, E.; Kamgar, K.; Yeh, C.-C.M.; Zhu, Y.; Gharghabi, S.; Ratanamahatana, C.A.; Chen, Y.; Hu, B.; Begum, N.; et al. The UCR Time Series Classification Archive. 2018. Available online: [https://www.cs.ucr.edu/~eamonn/time\\_series\\_data\\_2018/](https://www.cs.ucr.edu/~eamonn/time_series_data_2018/) (accessed on 31 May 2024).
89. Ratanamahatana, C.A.; Keogh, E. Everything you know about dynamic time warping is wrong. In Proceedings of the Third Workshop on Mining Temporal and Sequential Data, Seattle, WA, USA, 22 August 2004; Volume 32.
90. Kamper, S.J. Confidence Intervals: Linking Evidence to Practice. *J. Orthop. Sport. Phys. Ther.* **2019**, *49*, 763–764. [[CrossRef](#)] [[PubMed](#)]
91. Knowles, J.; Corne, D. On metrics for comparing nondominated sets. In Proceedings of the 2002 Congress on Evolutionary Computation (CEC’02), Honolulu, HI, USA, 12–17 May 2002; Volume 1, pp. 711–716. [[CrossRef](#)]
92. Ascia, G.; Catania, V.; Palesi, M. A GA-based design space exploration framework for parameterized system-on-a-chip platforms. *IEEE Trans. Evol. Comput.* **2004**, *8*, 329–346. [[CrossRef](#)]
93. Uddin, S.; Haque, I.; Lu, H.; Moni, M.A.; Gide, E. Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction. *Sci. Rep.* **2022**, *12*, 6256. [[CrossRef](#)] [[PubMed](#)]
94. Wiyono, S.; Abidin, T. Implementation of K-Nearest Neighbour (KNN) Algorithm to Predict Student’s Performance. *Simetris J. Tek. Mesin Elektro Dan Ilmu Komput.* **2018**, *9*, 873–878. [[CrossRef](#)]
95. Qian, Q.; Chen, S. Co-metric: A metric learning algorithm for data with multiple views. *Front. Comput. Sci.* **2013**, *7*, 359–369. [[CrossRef](#)]

**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.