Incremental Missing-Data Imputation for Evolving Fuzzy Granular Prediction

Cristiano Garcia, Daniel Leite, and Igor Škrjanc

***Abstract*—Missing values are common in real-world data stream applications. This article proposes a modified evolving granular fuzzy-rule-based model for function approximation and time-series prediction in an online context, where values may be missing. The fuzzy model is equipped with an incremental learning algorithm that simultaneously imputes missing data and adapts model parameters and structure over time. The evolving fuzzy granular predictor (eFGP) handles single and multiple missing values on data samples by developing reduced-term consequent polynomials and utilizing time-varying granules. Missing at random (MAR) and missing completely at random (MCAR) values in nonstationary data streams are approached. Experiments to predict monthly weather conditions, the number of bikes hired on a daily basis, and the sound pressure on an airfoil from incomplete data streams show the usefulness of eFGP models. Results were compared with those of state-of-the-art fuzzy and neuro-fuzzy evolving modeling methods. A statistical hypothesis test shows that eFGP outperforms other evolving intelligent methods in online MAR and MCAR settings, regardless of the application.**

***Index Terms*—Data stream, evolving intelligence, fuzzy system, incremental learning, missing-data imputation.**

I. Introduction

KNOWLEDGE discovery from data streams is helpful for many practical purposes. Detecting frequent patterns, trends, seasonalities, and nonstationarities may help human decision making in a variety of situations, applications, and endeavors. Data mining, machine learning, and computational intelligence methods have been applied to the purpose of finding useful information in sets of data [1]. These methods generally fit data patterns into classification or prediction models.

Data sources may be static or time-varying depending on whether their probability distributions change over time. In this context, *static* very often means that the source remains nearly unchanged so that data collected in a specific time interval are sufficient to represent reasonably well the behavior of the process or phenomenon in the future. Conversely, *time-varying* means that the data distribution is subject to variations (concept change), and online learning and model adaptation are necessary.

Models to deal with time-varying data streams must take into consideration the following.

1) Samples cannot be permanently stored.
2) Datasets are potentially unbounded.
3) Processing time should scale linearly (or at least polynomially) with the number of samples, attributes, and model parameters.
4) The data distribution may change gradually (concept drift) or abruptly (concept shift) at any time, or new patterns may emerge [1]–[12].

A survey on evolving fuzzy and neuro-fuzzy systems that deal with these fundamental issues and with some additional challenges of online-modeling and learning scenarios was recently published in [13].

The vast majority of stream-oriented learning methods require the values of all attributes to be available to work properly. However, missing values are common in real-world applications. Missing data arise due to incomplete observations, data transfer problems, malfunction of sensors or devices, incomplete information obtained from experts or on public surveys, among others [14], [15]. Three ways of treating missing data can be mentioned [16]:

1) discard samples, or even attributes, with many missing data;
2) impute values by maximum likelihood and parameter estimation procedures;
3) identify relationships among attributes and from previous values of an attribute to estimate new values.

Statistical and intelligent methods have been proposed to deal with missing data, especially in offline settings, where historical datasets are available [17], [18]. General methods entail deleting samples that contain one or more missing values, deleting attributes with more than a predefined percentage of their values missing, or imputing zeros, mean, or median for an attribute. After imputation, the complete, approximated, dataset is used for learning, classification, or prediction [15], [17], [19]. Model-based imputation methods provide a nonlinear way of handling missing data. These methods very often outperform general and independent methods [16], [20].

Manuscript received September 12, 2018; revised March 30, 2019 and August 7, 2019; accepted August 9, 2019. Date of publication August 15, 2019; date of current version October 6, 2020. The work of D. Leite was supported by the Serrapilheira Institute under Grant Serra-1812-26777. The work of I. Škrjanc was supported by the Slovenian Research Agency through Program P2-0219: Modeling, Simulation, and Control. *(Corresponding author: Daniel Leite.)*

C. Garcia and D. Leite are with the Department of Automatics, Federal University of Lavras, Lavras 37200-000, Brazil (e-mail: cristiano.garcia@ufla.br; daniel.leite@ufla.br).

I. Škrjanc is with the Faculty of Electrical Engineering, University of Ljubljana, 1000 Ljubljana, Slovenia (e-mail: igor.skrjanc@fe.uni-lj.si).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TFUZZ.2019.2935688

1063-6706 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

In [21], for example, several standalone imputation methods were overcome in a number of datasets and situations by imputation methods used in conjunction with a given predictive model.

Evolving systems are intelligent systems that, differently from adaptive and machine-learning systems, learn their parameters and structure simultaneously using a stream of data [1], [13], [22], [23]. The structural elements of evolving systems can be artificial neurons, fuzzy rules, data clusters, data clouds, subtrees, or information granules [3], [13], [23]. To deal with nonlinear and time-varying processes, evolving models should be updated through the use of online learning algorithms so that eventually large data flows can be processed in real time. The use of offline methods in this kind of problem is infeasible due to the unavailability of data to offline training, and tight time and memory constraints [1], [13]. In this article, missing values in nonstationary data streams is for the first time considered by means of an evolving approach.

An evolving fuzzy-rule-based model, modified to include an incremental learning algorithm for missing-data imputation and model adaptation, is proposed in this article. The result, which we call the evolving fuzzy granular predictor (eFGP), is useful for function approximation and prediction problems. Different from many statistical and machine-learning methods, the eFGP is suited for time-varying environments subject to concept drifts and shifts. Its rule base is built incrementally from scratch; meanwhile, the parameters of fuzzy granules and local linear functions are computed recursively based on data streams that may contain missing at random (MAR) and missing completely at random (MCAR) data. The eFGP handles single missing values on data samples by developing reduced-term consequent polynomials. Additionally, multiple missing values on samples are dealt with using the midpoints of time-varying granules evolved in the data space.

A granule is a clump of entities that may originate at the numeric or granular level and are arranged together due to their similarity, proximity, functionality, or coherency [1], [24]. Information granulation means that, instead of dealing with detailed real-world data, the data are considered in a more abstract and conceptual perspective. The goal of a time-varying granule is to catch the essence of the most recent data in a concise and explainable manner. Whenever yes-or-no quantification of concepts becomes too restrictive, fuzzy sets offer the feature of describing granules whose constituting elements may belong only partially. Fuzzy sets avoid specifying solid borders between full belongingness and full exclusion by means of smooth transition boundaries [24].

The remainder of this article is organized as follows. Section II presents essential notions of missing-data imputation and a literature review. Section III describes the eFGP approach, which differs from other evolving-intelligent approaches, because it addresses single and multiple MAR and MCAR values in data streams by means of reduced-term consequent polynomials and adaptive granules. Section IV presents prediction results, discussions, and comparisons to other evolving fuzzy and neuro-fuzzy methods. Section V concludes this article.

II. Contextualization of Model-Based Imputation Methods

A. Fundamental Concepts

Missing data is a common occurrence. A datum is missing if no value is available for the underlying attribute of a sample. There exist several ways to handle missing data. Some of the simplest ways are the following:

1) casewise deletion, where the whole observation is removed;
2) replacing the missing value with either zero or the mean for the attribute;
3) estimating the missing value using a special type of model, known as an *imputation model* [20].

The first two options may be used if the number of missing values is small and/or the dataset is large enough that a few lost or approximated samples are relatively insignificant [18], [20], [25]. Both case deletion and imputation transform an incomplete dataset into a fully populated rectangular format.

Depending on the nature of the missing data, they can be classified as follows.

1) *MCAR*: The propensity for a value to be missing is completely random, i.e., the probability of any value being missing is equal to the probability of any other value being missing.
2) *MAR*: The propensity of the values of a specific attribute to be missing is higher than those of other attributes, e.g., a sensor is not working properly, or one of the questions in a survey is harder to answer than the others.
3) *Not missing at random*: Missing values are dependent on other missing values, e.g., a sensor is not working properly in a portion of its range of values, or people tend not to answer a question in a survey if their depression level is high.

Slightly different definitions and interpretations may be found across research communities.

Data imputation should consider the nature of the data. For nominal attributes (words), for example, there are approaches such as the Global Most Common method [26]. For numerical data (real numbers), there exist a variety of simpler intuitive and model-based approaches [19], [25], [27].

A negative consequence of using simple imputation methods is that compelling a missing value to take the previously found value, the attributes mean value, or a value of zero may produce a dataset with distorted covariances and correlations. If this dataset is applied to a machine-learning system, an inaccurate model may be obtained leading to poor approximations or erroneous conclusions. Removing incomplete observations from the dataset may cause substantial loss of information depending on the fraction of incomplete samples.

Model-based imputation methods are grounded on statistical and machine-learning fundamentals. While linear and stationary interpolation for imputation may naturally make the completed dataset biased, depending on the level of nonlinearity and nonstationarity involved, nonlinear model-based imputation focuses on replacing the missing value with the best estimate, based on

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

previously sampled data [25]. Data streams impose further challenges to nonlinear modeling. Imputation and model adaptation should require reasonably limited time and memory. Incremental algorithms should scan samples only once [1], [2], [13].

The eFGP method described in the next section is suitable for online nonstationary environments. When the data stream changes (gradually or abruptly), a fuzzy model with reduced-term consequent polynomials—useful for prediction and data imputation—is adapted (parametrically or structurally). While most methods to deal with missing data cope with a single missing value per sample, the eFGP is robust to multiple MAR and MCAR values per data-stream sample.

B. Brief Literature Review

Intelligent methods for missing-data imputation usually rely on various fuzzy-clustering techniques, such as fuzzy c-means (FCM) [28]–[30], support vector regression (SVR) [28], [31], or neural networks [29], [32]. These techniques may be combined with machine-learning procedures. Often, metaheuristics, such as genetic algorithm (GA) [28] or other bioinspired techniques [32]–[34], are used for parameter adjustment. Intelligent models provide a nonlinear way to impute values based on information uncovered from the data.

A method to integrate SVR, FCM, and GA is proposed in [28]. FCM and SVR are trained with "complete" data, i.e., with samples without missing values. Samples with missing values are used as inputs to a fuzzy model (using partial distance calculation), and to the trained SVR model. Estimations by both models are compared. If their difference is greater than a predefined error threshold, then the GA is used to adapt the number of fuzzy clusters and an FCM weighting factor. Otherwise, FCM is used to estimate the missing values. Results were compared with an imputation-with-zeros approach and with other model combinations such as FCM-GA and SVR-GA. The proposed SVR-FCM-GA approach provided more accurate results.

A missing-data-estimation scheme, similar to [28], is discussed in [14]. FCM and SVR are replaced by a weighted K-nearest neighbor algorithm and an autoassociative neural network known as autoencoder. Both [14] and [28] assume MAR type of data and, therefore, depend on the correlation between attributes. Maximum likelihood procedures are also commonly used for data imputation. Bánbura and Modugno [35] point out that maximum likelihood has a number of advantages for small datasets with different patterns, nonlinear dynamics, and higher proportions of missing values.

A review on missing-data imputation methods for offline classification is given in [17]. A number of imputation algorithms such as regularized expectation–maximization, singular value decomposition, and the Bayesian principal component analysis [36] were analyzed. Empirical results obtained by applying the Wilcoxon signed rank test strongly argued in favor of the convenience of using imputation methods instead of resorting to case deletion or lack of imputation. However, there is no consensus about a best method for different datasets. Handling missing values has also been considered in conjunction with other pattern classification topics such as imbalanced datasets, semisupervised learning, scalability, and temporal databases [36]–[38].

Grounded in the missing-data theory and well-validated statistical principles, two flagship techniques in modern missing-data analysis have shown to be promising [39]. The first concerns model-based or randomization-based inference for multiple imputation. The second is full information maximum likelihood (FIML) analyses. FIML requires a set of auxiliary variables, correlated with variables containing missingness, to be included in the model to avoid biased estimates [40]. Randomization-based inference requires repeatedly chosen samples under a specific experiment. Therefore, FIML and random inference cannot be directly applied to nonstationary data streams, since the data are not available *a priori* for correlation analyses and resampling, and their statistical properties change over time. Evolving model-based inference for multiple imputation and prediction in online nonstationary environment is investigated in this article.

Limitations of the aforementioned methods include the following.

1) Prior availability of a closed input dataset is generally necessary.
2) Offline learning methods require multiple passes over the data to determine the parameters of the imputation model, in spite of the use of nonlinear models.
3) The resulting classifier or predictor does not deal with the types of changes typical of nonstationary environments.

Evolving methods, such as the eFGP, presented in this article, address these limitations and are appropriate to nonstationary environments subject to concept changes. eFGP self-develops its rule structure and updates its parameters to handle, respectively, abrupt and gradual changes in online data streams. This characteristic is a key advantage of the eFGP over the majority of machine learning and computational intelligence methods.

The fundamental idea is that the eFGP's learning updates the parameters of a fuzzy granule that is eventually located closer to the current data sample compared to the other granules. If a sample does not belong to any granule, but belongs to the expansion region of some granules, then the closest granule is chosen to be updated. This granule expands its bounds to cover the sample, and at the same time, the coefficients of associated local linear functions are updated. The granule may be dragged toward new samples if such samples belong to the same region. Concept shift and new behaviors may produce samples located relatively far from the current granules. As these samples are not in the expansion region of any granule, it is not practical to drag a granule toward them because the pattern that such granule currently represents would be forgotten. Thus, the eFGP approach is to create a new granule to cover the region where such "far samples" are placed. This new granule is governed by a new rule, which expands the eFGP rule base.

III. Evolving Fuzzy Granular Predictor

A. Preliminaries

Evolving fuzzy-set-based granular modeling was proposed in [41] and [42] as a framework for modeling data streams, in which the measured values are inherently uncertain. Information based on perception can be represented as fuzzy, interval, or numerical data to be taken into account in the fuzzy granular framework.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

TABLE I
LIST OF SYMBOLS AND DEFINITIONS

Symbol	Definition
$\mathbf{x}$	n-dimensional input vector
x_j	j-th attribute of the input vector
y	Actual output
$\hat{y}$	Estimated output
$[h]$	Time index (superscript)
γ^i	i-th information granule
x_θ	Missing value
$\|\theta\|$	Amount of missing values in a given $\mathbf{x}$
A_j^i	i-th trapezoidal membership function of the j-th input attribute
$\underline{\underline{a}}_j^i$	Lower limit of the support of A_j^i
$\underline{a}_j^i$	Lower limit of the core of A_j^i
$\overline{a}_j^i$	Upper limit of the core of A_j^i
$\overline{\overline{a}}_j^i$	Upper limit of the support of A_j^i
ρ_j	Maximum length that A_j^i can assume
B^i	i-th membership function of the output y
$\underline{\underline{b}}^i$	Lower limit of the support of B^i
$\underline{b}^i$	Lower limit of the core of B^i
$\overline{b}^i$	Upper limit of the core of B^i
$\overline{\overline{b}}_j^i$	Upper limit of the support of B^i
σ	Maximum length that B^i can assume
p^i	Complete affine consequent function of the i-th rule
α_j^i	j-th coefficient of function p^i
q_θ^i	θ-th consequent function with reduced argument-list of the i-th rule – the θ-th term is omitted
$\beta_{j\theta}^i$	j-th coefficient of function q_θ^i
Ψ_{com}^i	Activation degree of the i-th rule for the complete function p^i
Ψ_{inc}^i	Activation degree of the i-th rule for an incomplete function q_θ^i
ch	Convex hull operator
mp	Midpoint operator
h_r	Number of iterations to perform merging and deleting
RMSE	Root Mean Square Error
NDE	Non-Dimensional Error

Stream data are compressed to a few granules whose location and granularity reflect the structure of the data. This article aims to model and process numerical (pointwise) data only. We focus on the question of missing data and model robustness.

The eFGP provides pointwise and granular prediction of nonstationary functions and linguistic description of the behavior of a system. Local eFGP models consist of IF–THEN rules developed incrementally from the data. Learning can start from an empty rule base, and as new information arises, granules and rules are created and their parameters updated over time. The eFGP is, therefore, inherently accommodating of change in the source data, so that the resulting models never need to be redesigned or retrained from scratch.

For each granule constructed in the data space, there exists a corresponding rule that summarizes the general behavior of the elements that constitute the granule. The eFGP learning algorithm incrementally adapts granules and other parameters associated to rules so that the fuzzy model captures recent occurrences without forgetting previous behaviors.

Symbol conventions and definitions used in this article are summarized in Table I for reference.

B. Evolving Fuzzy Granular Model for Prediction and Missing-Data Imputation in an Online Environment

Let $(\mathbf{x}, y)^{[h]}$, $h = 1, \ldots$, be the hth observation of a data stream. $\mathbf{x} \in \Re^n$ is an input multidimensional vector, and $y \in \Re$ is the actual output. Vectors are denoted by boldface lowercase letters. The actual output $y^{[h]}$ will be known after the input $\mathbf{x}^{[h]}$ arrives and a prediction $\hat{y}^{[h]}$ is given. An attribute x_j of $\mathbf{x} = (x_1, \ldots, x_n)$ is a real value. The same holds for y. The pair $(\mathbf{x}, y)$ is a point in the product space $X \times Y$. Let $\gamma^i \in X \times Y$, $i = 1, \ldots, c$, be the current set of eFGP granules built on the basis of $(\mathbf{x}, y)$.

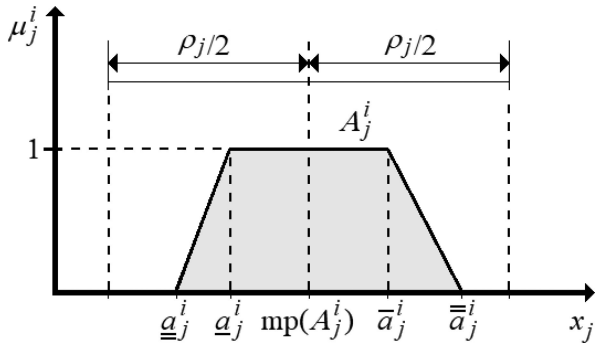

Fig. 1. Trapezoidal membership function.

Rules R^i governing granules γ^i are given as

$$R^i\text{: IF } (x_1 \text{ is } A_1^i) \text{ AND ...AND } (x_n \text{ is } A_n^i),$$
$$\text{THEN } \underbrace{(y \text{ is } B^i)}_{\text{Linguistic}} \text{ AND } \Bigg(\underbrace{\hat{y} = p^i(x_1, \ldots, x_n)}_{\text{Functional}}$$
$$\text{OR } \underbrace{\hat{y} = q_\theta^i(x_1, \ldots, x_{\theta-1}, x_{\theta+1}, \ldots, x_n),\ \theta = 1, \ldots, n}_{\text{Functional with reduced argument-list } (x_\theta \text{ omitted})} \Bigg)$$

where x_θ is a missing value; p^i and q_k^i are affine functions; and $A_j^i = (\underline{\underline{a}}_j^i, \underline{a}_j^i, \overline{a}_j^i, \overline{\overline{a}}_j^i)$ and $B^i = (\underline{\underline{b}}^i, \underline{b}^i, \overline{b}^i, \overline{\overline{b}}^i)$ are trapezoidal membership functions related to the ith rule. Trapezoidal functions are canonically represented by four parameters listed in ascending order (see Fig. 1). The intermediate parameters of A_j^i and B^i form the core. The core of a membership function, say A_j^i, is the region $[\underline{a}_j^i, \overline{a}_j^i]$ of the universe (range of possible values) of x_j characterized by elements with full membership in the set A_j^i. The boundary parameters of a membership function form its support. The support of A_j^i is the region $[\underline{\underline{a}}_j^i, \overline{\overline{a}}_j^i]$ of the universe x_j characterized by elements with nonzero membership in the set A_j^i. The membership degree of x_j in A_j^i is given by μ_j^i. Therefore, if x_j belongs to the core of A_j^i, then $\mu_j^i = 1$. If x_j does not belong to the support of A_j^i, then $\mu_j^i = 0$. Notice that q_k^i has one term less than p^i and that a disjunction operator (OR) relates the terms. The set of rules R^i, $i = 1, \ldots, c$, is a fuzzy granular description of a system. Initially, $c = 0$, i.e., no prior knowledge is assumed. A rule provides a granular (by means of active output fuzzy sets B^i) and a pointwise (by means of p^i or q_θ^i) prediction. The functional consequent is given by either p^i, in case $\mathbf{x}^{[h]}$ is complete, or q_θ^i, in case x_θ is missing. The linguistic consequent offers prediction bounds and interpretability since trapezoids B^i can be connected to linguistic values.

Affine functions are given as

$$p^i(x) = \alpha_0^i + \sum_{j=1}^{n} \alpha_j^i x_j \tag{1}$$

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

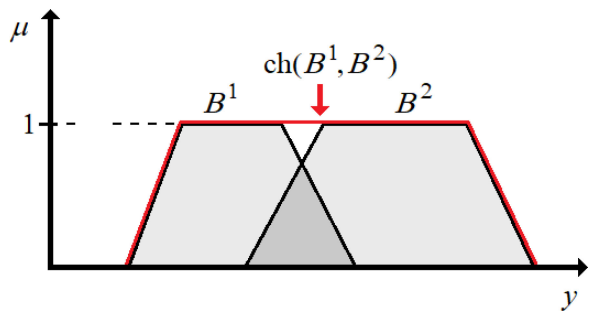

Fig. 2. Example of convex hull of the membership functions B^1 and B^2 defined in the universe of the output variable y.

and

$$q^i_\theta(x) = \beta^i_{0\theta} + \sum_{j=1, j\neq\theta}^{n} \beta^i_{j\theta} x_j \tag{2}$$

$\theta = 1, \ldots, n$. In general, coefficients α^i_j and $\beta^i_{j\theta}$, $j = 0, 1, \ldots, n; j \neq \theta$, are not linearly correlated in the sense of Pearson. They can only be fully correlated if the coefficient α^i_θ, directly related to a particular missing value x_θ, is equal to 0. Consequent functions, p^i and $q^i_\theta\ \forall\theta$, are updated using the recursive least-squares algorithm [42]. Subsequently, Section III-C addresses model structure and parameter adaptation.

As trapezoids A^i_j may overlap, eFGP pointwise prediction is found as the weighted mean value

$$\hat{y} = \frac{\sum_{i=1}^{c} \Psi^i_{\text{com}}\, p^i(x_1, \ldots, x_n)}{\sum_{i=1}^{c} \Psi^i_{\text{com}}} \tag{3}$$

for a complete $\mathbf{x}$, or

$$\hat{y} = \frac{\sum_{i=1}^{c} \Psi^i_{\text{inc}}\, q^i_\theta(x_1, \ldots, x_{\theta-1}, x_{\theta+1}, \ldots, x_n)}{\sum_{i=1}^{c} \Psi^i_{\text{inc}}} \tag{4}$$

if x_θ is missing. The activation degree of the ith rule for $\mathbf{x}$ complete or incomplete, Ψ^i_{com} or Ψ^i_{inc}, is obtained from

$$\Psi^i_{\text{com}} = T(A^i_1(x_1), \ldots, A^i_n(x_n)) \tag{5}$$

or

$$\Psi^i_{\text{inc}} = T(A^i_1(x_1), \ldots, A^i_{\theta-1}(x_{\theta-1}), A^i_{\theta+1}(x_{\theta+1}), \ldots \\ \ldots, A^i_n(x_n)). \tag{6}$$

T is any triangular norm. The minimum (Gödel) T-norm is used in this article, but other choices are possible.

Granular prediction is given by the convex hull of sets B^{i^*}, where i^* are indices of active granules for $\mathbf{x}^{[h]}$. The convex hull of a set of trapezoids, say $B^1, \ldots, B^c$, is given as

$$\text{ch}(B^1, \ldots, B^c) = (T(\underline{b}^1, \ldots, \underline{b}^c), T(\underline{b}^1, \ldots, \underline{b}^c), \\ S(\overline{b}^1, \ldots, \overline{b}^c), S(\overline{\overline{b}}^1, \ldots, \overline{\overline{b}}^c)) \tag{7}$$

where S is the selected T-norm's corresponding conorm (the maximum T-conorm in this case). Fig. 2 illustrates the convex hull operation of the trapezoids B^1 and B^2 defined in the output dimension. The result is also a trapezoidal membership function. Trapezoidal prediction given by ch(.) encloses $\hat{y}$ and may help decision making and improve model acceptability. For example, the enclosure of $\hat{y}$, provided by the support of active trapezoids, may be interpreted as optimistic and pessimistic estimates in an application. Often, such enclosure of the output data can be more important than the numerical estimates, since, with pointwise values, we have no idea about the error or uncertainty involved in the estimation.

Suppose a sample $\mathbf{x}^{[h]}$ has multiple missing values, say x_{θ_1} and x_{θ_2}. A straightforward, but not practical, approach to deal with multiple missing data is to consider additional consequent functions with fewer terms. However, the number of parameters to be updated would scale exponentially with the number of attributes.

An effective approach for multiple missing values we employ for eFGP modeling consists of imputing the midpoint of the input membership functions related to the most active rule for the missing values. In this case, the activation level of the rule R^i, $i = 1, \ldots, c$, is calculated as

$$\Psi^i_{\text{inc}} = T(A^i_1(x_1), \ldots, A^i_{\theta_1-1}(x_{\theta_1-1}), A^i_{\theta_1+1}(x_{\theta_1+1}), \\ \ldots, A^i_{\theta_2-1}(x_{\theta_2-1}), A^i_{\theta_2+1}(x_{\theta_2+1}), \ldots, A^i_n(x_n)). \tag{8}$$

If R^i is the most active rule for $\mathbf{x}^{[h]}$ according to Ψ^i_{inc}, then the midpoint of its membership functions related to the missing values is used for imputation. The midpoint is the mean value of the core parameters of the membership function, i.e.,

$$x^{[h]}_{\theta_1} = \text{mp}(A^i_{\theta_1}) = \frac{(\underline{a}^i_{\theta_1} + \overline{a}^i_{\theta_1})}{2} \tag{9}$$

and

$$x^{[h]}_{\theta_2} = \text{mp}(A^i_{\theta_2}) = \frac{(\underline{a}^i_{\theta_2} + \overline{a}^i_{\theta_2})}{2}. \tag{10}$$

The imputed (complete) sample is used by the fuzzy model to provide numerical and granular predictions at the time step h. If $\mathbf{x}^{[h]}$ has multiple missing values, (8) is used to choose the most active rule. After the imputation from (9) and (10), the complete equations, (1), (3), and (5), are combined to produce the numerical prediction. Additionally, the support of the convex hull of active membership functions, obtained from (7), is used to provide the granular prediction. The multiple-imputation procedure extends straightforwardly to larger amounts of missing data per sample.

C. Incremental Adaptation of the Fuzzy Granular Model

Let the midpoint of a trapezoidal membership function be the average of its core parameters, similar to (9). Moreover, let ρ_j and σ be the maximum length a granule can assume, respectively, along the jth input dimension and output dimension. Parameter ρ_j delimits the maximum expansion region of trapezoidal membership functions around their midpoints (see Fig. 1). In other words, the parameters of a membership function, A^i_j, must not assume values lower than $\text{mp}(A^i_j) - \rho_j/2$ nor values greater than $\text{mp}(A^i_j) + \rho_j/2$ at any time step. Different values of ρ_j produce different representations of the same data set in different

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

levels of granularity. $\rho_j\ \forall j$ and σ assume a single value in $[0, 1]$. If ρ_j is equal to 0, then granules are not expanded. Learning creates a new rule for each sample, which causes overfitting. If ρ_j is equal to 1, then a single granule covers the entire data domain. Evolvability is reached by choosing intermediate values for ρ_j. The higher the value of ρ_j, the more compact tends to be the structure of the resulting eFGP model.

A new granule, γ^{c+1}, is created by adding a rule, R^{c+1}, to the current set of rules, $R = \{R^1, \ldots, R^c\}$. Granule and rule are created whenever either an input vector, $\mathbf{x}^{[h]}$, contains at least one element, $x_j^{[h]}$, $j = 1, \ldots, n$, that is not in the expansion region of A_j^i, $i = 1, \ldots, c$, or $y^{[h]}$ is not in the expansion region of B^i, $i = 1, \ldots, c$. Formally, $x_j^{[h]}$ must belong to $[\text{mp}(A_j^i) - \rho/2, \text{mp}(A_j^i) + \rho/2]$, $j = 1, \ldots, n$, to be considered by the ith granule. Additionally, the actual output, $y^{[h]}$, must belong to $[\text{mp}(B^i) - \sigma/2, \text{mp}(B^i) + \sigma/2]$. Notice that $A_j^i\ \forall j$ and B^i have their parameters updated if 1) data samples are in their expansion regions, and 2) R^i is the most active rule.

The new granule, γ^{c+1}, has trapezoidal memberships functions, A_j^{c+1} and B^{c+1}, in which

$$\underline{\underline{a}}_j^{c+1} = \underline{a}_j^{c+1} = \overline{a}_j^{c+1} = \overline{\overline{a}}_j^{c+1} = x_j^{[h]}, \qquad j = 1, \ldots, n \tag{11}$$

and

$$\underline{\underline{b}}^{c+1} = \underline{b}^{c+1} = \overline{b}^{c+1} = \overline{\overline{b}}^{c+1} = y^{[h]}. \tag{12}$$

Therefore, the new granule, γ^{c+1}, is initially a point in the data space, the point $(\mathbf{x}, y)$. The coefficients of the complete consequent function, p^{c+1}, are

$$\alpha_j^{c+1} = 0,\ j \neq 0, \text{ and } \alpha_0^{c+1} = y^{[h]}. \tag{13}$$

Similarly, the coefficients of the incomplete functions, $q_\theta^{c+1}\ \forall\theta$, are initialized as

$$\beta_{j\theta}^{c+1} = 0,\ j \neq 0, \text{ and } \beta_{0\theta}^{c+1} = y^{[h]}. \tag{14}$$

The adaptation of a rule R^i consists of: 1) expanding or contracting the support and the core of $A_j^i\ \forall j$ and B^i to accommodate new data; and 2) updating the coefficients of complete and incomplete consequent functions, p^i and $q_\theta^i \forall\theta$.

If a data sample, $(\mathbf{x}, y)^{[h]}$, belongs to the expansion region of a granule, γ^i, then its membership functions are enlarged to cover the sample. If the sample is within γ^i, its parameters can be changed in the sense of contracting or expanding the core of its membership functions. The following situations may happen according to the position of a sample in relation to a granule (refer to Fig. 1).

1) If $x_j^{[h]} \in [\text{mp}(A_j^i) - \frac{\rho_j}{2}, \underline{\underline{a}}_j^i]$,
   then $\underline{\underline{a}}_j^i(\text{new}) = x_j^{[h]}$ (support expansion).
2) If $x_j^{[h]} \in [\underline{\underline{a}}_j^i, \underline{a}_j^i]$,
   then $\underline{a}_j^i(\text{new}) = x_j^{[h]}$ (core expansion).
3) If $x_j^{[h]} \in [\underline{a}_j^i, \text{mp}(A_j^i)]$,
   then $\underline{a}_j^i(\text{new}) = x_j^{[h]}$ (core contraction).
4) If $x_j^{[h]} \in [\text{mp}(A_j^i), \overline{a}_j^i]$,
   then $\overline{a}_j^i(\text{new}) = x_j^{[h]}$ (core contraction).
5) If $x_j^{[h]} \in [\overline{a}_j^i, \overline{\overline{a}}_j^i]$,
   then $\overline{a}_j^i(\text{new}) = x_j^{[h]}$ (core expansion).
6) If $x_j^{[h]} \in [\overline{\overline{a}}_j^i, \text{mp}(A_j^i) + \frac{\rho_j}{2}]$,
   then $\overline{\overline{a}}_j^i(\text{new}) = x_j^{[h]}$ (support expansion). (15)

When operating on core parameters, $\underline{a}_j^i$ and $\overline{a}_j^i$, adjustment of the midpoint of γ^i is required. Thus, we have

$$\text{mp}(A_j^i)(\text{new}) = \frac{\underline{a}_j^i(\text{new}) + \overline{a}_j^i(\text{new})}{2} \tag{16}$$

$j = 1, \ldots, n$. Support contraction may be necessary as a consequence of the midpoint adaptation. Thus, we have the following.

1) If $\text{mp}(A_j^i)(\text{new}) - \frac{\rho_j}{2} > \underline{\underline{a}}_j^i$,
   then $\underline{\underline{a}}_j^i(\text{new}) = \text{mp}(A_j^i)(\text{new}) - \frac{\rho_j}{2}$.
2) If $\text{mp}(A_j^i)(\text{new}) + \frac{\rho_j}{2} < \overline{\overline{a}}_j^i$,
   then $\overline{\overline{a}}_j^i(\text{new}) = \text{mp}(A_j^i)(\text{new}) + \frac{\rho_j}{2}$. (17)

Adaptation of output trapezoids, B^i, uses data $y^{[h]}$ and relations analogous to those of (15)–(17). Only the most active granule, γ^i, is chosen to be adapted for a sample $(\mathbf{x}, y)^{[h]}$.

All consequent functions, p^i and $q_\theta^i\ \forall\theta$, are updated using the recursive least squares algorithm [42] in case R^i is the most active rule for a complete $\mathbf{x}^{[h]}$. However, for a complete $\mathbf{x}^{[h]}$, coefficients $\beta_{j\theta}^i$, $j = 0, 1, \ldots, n$; $j \neq \theta$, are computed ignoring the attribute $x_\theta^{[h]}$, $\theta = 1, \ldots, n$. For an incomplete $\mathbf{x}^{[h]}$, with a unique missing element, $x_\theta^{[h]}$, only the coefficients of q_θ^i are updated. In case $\mathbf{x}^{[h]}$ contains multiple missing values, consequent coefficients are not updated.

After a number of time steps, h_r, merging and deleting rules may help to keep the fuzzy model succinct and updated. Automatic procedures to merge and delete rules are described as follows.

Granules may drift in the data space and become too close to each other. In this case, they may represent a single pattern so that merging them is important to keep a compact rule base. The granule that results from merging must respect the limits imposed by ρ_j and σ. Neighbor granules, say γ^1 and γ^2, are merged into a single granule, γ^Ψ, by combining their trapezoidal membership functions through the convex hull operator. Formally, we have

$$\begin{aligned} A_j^\Psi &= \text{ch}(A_j^1, A_j^2) \\ &= (T(\underline{\underline{a}}_j^1, \underline{\underline{a}}_j^2), T(\underline{a}_j^1, \underline{a}_j^2), S(\overline{a}_j^1, \overline{a}_j^2), S(\overline{\overline{a}}_j^1, \overline{\overline{a}}_j^2)) \end{aligned} \tag{18}$$

$j = 1, \ldots, n$, and

$$\begin{aligned} B^\Psi &= \text{ch}(B^1, B^2) \\ &= (T(\underline{\underline{b}}^1, \underline{\underline{b}}^2), T(\underline{b}^1, \underline{b}^2), S(\overline{b}^1, \overline{b}^2), S(\overline{\overline{b}}^1, \overline{\overline{b}}^2)). \end{aligned} \tag{19}$$

Parameters of consequent functions of merged rules are obtained from

$$\alpha_j^\Psi = \frac{\alpha_j^1 + \alpha_j^2}{2},\ j = 0, \ldots, n \tag{20}$$

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

and

$$\beta_{j\theta}^{\Psi} = \frac{\beta_{j\theta}^{1} + \beta_{j\theta}^{2}}{2},\ j = 0, \ldots, \theta-1, \theta+1, \ldots, n;\ \forall\theta. \quad (21)$$

This strategy helps to reduce the number of rules and overlapped granules, covering similar information.

Concept change may cause granules and rules to become inactive. In this case, the most recent data do not fall in the region that a given granule covers for a number of iterations, i.e., in the region where past data used to fall. The rule that governs the granule becomes useless in the current context. Rules and granules are removed if they are not activated during h_r time steps. The purpose is to keep the fuzzy rule base concise and formed only by elements that are useful to deal with the current environment. Nevertheless, some applications may require memorization of old and outdated events. In this case, parameter h_r may be set to ∞ to preserve the rules forever. The value chosen for h_r depends on how long we want to keep inactive rules in the memory of the model. Both eFGP parameters, ρ_j and h_r, depend on the purpose of the model. Different from ρ_j, parameter h_r has a secondary influence on the prediction performance.

The learning procedure to evolve an eFGP model is summarized in Algorithm 1. Steps 3 and 22 emphasize that samples are received and deleted one at a time—an essential feature in data stream processing. $|\theta|$ denotes the number of missing elements in a given $\mathbf{x}^{[h]}$. The resulting eFGP model is available at any time. The model is robust to single and multiple missing values per sample using a modified rule structure and an inherent (nonlinear and nonstationary) mechanism of learning and data imputation.

Algorithm 1: eFGP Online Incremental Learning.

1: Define ρ, σ and h_r;
2: **while** 1 **do**
3: Read input $\mathbf{x}^{[h]}, h = 1, \ldots$;
4: //Prediction
5: **if** $|\theta| = 0$ **then**
6: Give prediction $\hat{y}$ using complete functions p^i [see (1), (3), and (5)];
7: **else if** $|\theta| = 1$ **then**
8: Give prediction $\hat{y}$ using reduced-term functions q_θ^i [see (2), (4), and (6)];
9: **else if** $|\theta| > 1$ **then**
10: Choose the most active rule for $\mathbf{x}^{[h]}$ [see (8)];
11: Multiple imputation using the midpoints of the most active granule for $\mathbf{x}^{[h]}$ [see (9) and (10)];
12: Use complete functions p^i to give the prediction $\hat{y}$ [see (1), (3), and (5)];
13: **end if**
14: Provide the granular prediction [see (7)];
15: //Model Adaptation ($y^{[h]}$ becomes available)
16: **if** ($x_j^{[h]} \notin [\text{mp}(A_j^i) - \rho/2, \text{mp}(A_j^i) + \rho/2]$, $j = 1, \ldots, n$, OR $y^{[h]} \notin [\text{mp}(B^i) - \sigma/2, \text{mp}(B^i) + \sigma/2]) \,\forall i$ **then**
17: Create rule to accommodate $(\mathbf{x}, y)^{[h]}$ [see (11)–(14)];
18: **else**
19: Adapt the most active granule [see (15)–(17)];
20: Adapt consequent coefficients of the most active rule (recursive least squares [42]);
21: **end if**
22: Delete sample $(\mathbf{x}, y)^{[h]}$;
23: **if** $h = zh_r$, $z = 1, \ldots$ **then**
24: Delete inactive granules and rules;
25: Merge neighbor granules [see (18)–(21)];
26: **end if**
27: **end while**

IV. Results and Discussions

A. Datasets and Evaluation Metrics

Benchmark datasets were chosen to evaluate the efficiency of the eFGP. The datasets contain no missing data in principle, which is convenient for the purpose of the experiments and comparative analyses. They are the following.

1) Death Valley (Furnace Creek) weather dataset:[1] Records of monthly mean temperature in degrees Celsius from 1901 to 2009 (1306 observations) are considered. A fixed time window of 12 months, with no exogenous inputs, is used for one-step prediction. In Death Valley, superheated moving air masses are trapped by surrounding steep mountain ranges creating an extremely dry climate with high temperatures. Roof and Callagan [43] give a complete list of factors that produce high air temperatures and temperature variations in Death Valley.
2) Capital Bike Sharing dataset:[2] Located in Washington, DC, USA, this bike loan system contains two-year information (731 samples) of usage log data. Nine attributes are used: season, month, holiday, weekday, weather situation, temperature, apparent temperature, air humidity, and wind speed. The count of bikes hired in a day is the output. Sharing systems are a new way of renting bikes. Users can take and return a bike in different positions of a city. Apart from many applications of these systems, the characteristics of real-time data—duration of travel, departure and arrival position, and others—are attractive as a virtual sensor network that can be used to analyze mobility. Important events in a city can be detected by monitoring these data. Bike sharing plays an important role in traffic, environmental, and health issues.
3) Airfoil Self-Noise dataset:[3] This dataset is related to a series of aerodynamic and acoustic tests conducted in an anechoic wind tunnel at the National Aeronautics and Space Administration using different NACA 0012 airfoils. This is a regression problem that contains 1503 samples;

[1][Online]. Available: https://www.nps.gov/deva/planyourvisit/weather.htm

[2][Online]. Available: https://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset

[3][Online]. Available: https://archive.ics.uci.edu/ml/datasets/airfoil+self-noise

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

five attributes, namely, frequency (Hertz), angle of attack (degree), chord length (meter), free-stream velocity (meters per second), and suction-side displacement thickness (meter); and an output, the scaled sound pressure (decibel). Airfoil self-noise is due to the interaction between an airfoil blade and the turbulence produced in its own boundary layer and near wake [44]. It is the total noise produced when an airfoil encounters smooth nonturbulent inflow. The interest behind developing fundamental understanding and prediction models of the various self-noise mechanisms is motivated by its importance to broadband helicopter rotor, wind turbine, and airframe noises.

Prediction performance using the complete datasets and MCAR and MAR data were assessed. For MCAR scenarios, the chance of occurrence of missing values is equal among the attributes. eFGP models were constructed and analyzed using datasets containing from 0% to 30% of missing values—a range usually used in related studies [28], [29], [31]. For example, for a dataset with 1000 entries, a random number between 1 and 1000 is chosen. After deleting the selected entry, a random number between 1 and 999 is chosen, and so on until a given percentage of values is missing.

For MAR data, a specific attribute is more inclined to receive an empty reading in relation to the other attributes. In this case, an attribute is taken at random to have its values more likely to be missing. Prediction performance is evaluated for different percentages of chance that the chosen attribute and the rest of the attributes are missing. The cases are: 1) 5%–1% (which means 5% of chance that the value of the chosen attribute is missing, and 1% of chance that each of the remaining values is missing); 2) 10%–1%; 3) 10%–5%; 4) 20%–5%; 5) 20%–10%; 6) 30%–5%; and 7) 30%–10%. For example, for the 5%–1% case, and a dataset with ten attributes and 100 samples—a total of 1000 entries—a random number between 1 and 1400 is chosen. Each of the 100 entries related to the chosen attribute receives five numbers in the lottery, instead of a single number of the entries related to the other attributes.

Model accuracy is quantified using the RMSE index given as

$$\text{RMSE} = \frac{1}{H}\sum_{h=1}^{H}\sqrt{(\hat{y}^{[h]} - y^{[h]})^2} \tag{22}$$

where H is the number of iterations. The NDE index

$$\text{NDE} = \frac{\text{RMSE}}{\text{std}(y^{[h]}\forall h)} \tag{23}$$

where std(.) is the standard deviation, is useful to compare the accuracy of a predictor in different data streams.

Additionally, the eFGP provides granular outputs. The enclosure of the numerical data given by eFGP linguistic consequents, B^i, may be as important as pointwise estimates, $\hat{y}$, to assist decision making in an application. Therefore, we address numerical prediction accuracy, enclosures, and linguistic descriptions in the experiments.

Alternative evolving methods, namely, evolving granular neural network (eGNN) [45], evolving Takagi–Sugeno (eTS) [46], extended Takagi–Sugeno (xTS) [47], and fuzzy-set-based evolving modeling (FBeM) [42], are also evaluated either assuming that the datasets are complete or removing a fraction of incomplete samples and replicating the last output.

TABLE II
EFGP RESULTS FOR THE DEATH VALLEY WEATHER STATION ASSUMING MISSING DATA OF THE MCAR TYPE

MCAR	RMSE	NDE	Mean # of rules
0%	0.0579 +/- 0.0018	0.2239 +/- 0.0070	13.3 +/- 0.1
1%	0.0600 +/- 0.0014	0.2322 +/- 0.0055	13.9 +/- 0.5
5%	0.0636 +/- 0.0018	0.2461 +/- 0.0072	20.5 +/- 0.9
10%	0.0640 +/- 0.0020	0.2474 +/- 0.0080	22.9 +/- 0.8
15%	0.0703 +/- 0.0041	0.2719 +/- 0.0158	27.7 +/- 1.1
20%	0.0818 +/- 0.0059	0.3163 +/- 0.0223	28.8 +/- 1.1
30%	0.1142 +/- 0.0057	0.4180 +/- 0.0222	28.1 +/- 2.3

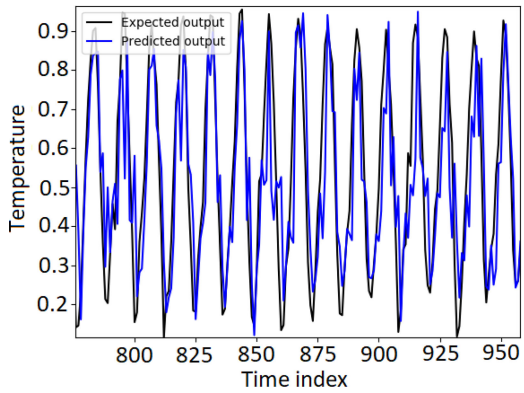

Fig. 3. Pointwise eFGP prediction for the mean monthly temperature of Death Valley considering 30% of MCAR values.

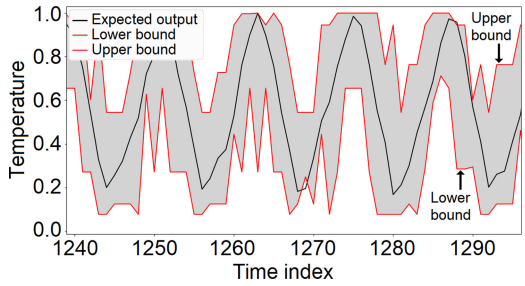

Fig. 4. Granular eFGP estimation for the Death Valley monthly mean temperature considering 30% of MCAR values.

B. *eFGP Results for MCAR Data*

Average results for the Death Valley dataset considering different fractions of MCAR values and ten runs of the eFGP learning algorithm for each case are shown in Table II. The eFGP initial parameters are $\rho = \sigma = 0.3$, and $h_r = 150$. As the percentage of missing data increases, the error indices increase monotonically, and the number of rules in the model structure tends to increase. New granules and rules are needed to cover incomplete samples, since correlation information is partially lost. As shown in Fig. 3, the eFGP monthly temperature pointwise estimates for the roughest, 30% MCAR scenario, track the peaks and valleys of the time series with reasonable accuracy. Similarly, Fig. 4 shows part of the granular prediction provided by the eFGP (for a clearer view). Notice that the bounds of the

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

granular prediction given by the support of active trapezoidal membership functions in the output domain enclose the actual temperature. If lower values for the granularity, ρ and σ, are chosen, then a narrower envelope can be achieved at the price of additional fuzzy rules. A tradeoff between model compactness, interpretability of rules, pointwise accuracy, and narrowness of the granular output should be evaluated depending on the purpose of the model.

eFGP rules can be accessed anytime. For example, an active rule at $h = 1295$ is

R^i: IF (x_1 is [0.54, 0.59, 0.70, 0.79]) AND (x_2 is [0.68, 0.68, 0.83, 0.87]) AND (x_3 is [0.82, 0.83, 0.87, 0.95]) AND (x_4 is [0.78, 0.84, 0.89, 0.98]) AND (x_5 is [0.73, 0.75, 0.80, 0.86]) AND (x_6 is [0.57, 0.64, 0.76, 0.77]) AND (x_7 is [0.49, 0.55, 0.61, 0.61]) AND (x_8 is [0.19, 0.20, 0.34, 0.40]) AND (x_9 is [0.15, 0.19, 0.25, 0.26]) AND (x_{10} is [0.16, 0.22, 0.32, 0.39]) AND (x_{11} is [0.27, 0.34, 0.43, 0.47]) AND (x_{12} is [0.39, 0.39, 0.51, 0.61])

THEN (y is $[0.51, 0.56, 0.72, 0.74]$) AND ($\hat{y}=0.14+0.17x_1 - 0.45x_2 + 0.64x_3 + 0.27x_4 - 0.28x_5 + 0.07x_6 - 0.28x_7 + 0.14x_8 - 0.09x_9 + 0.58x_{10} - 0.33x_{11} + 0.39x_{12}$ OR $\hat{y}=0.36 - 0.44x_2 + 0.55x_3 + 0.58x_4 - 0.43x_5 - 0.07x_6 - 0.24x_7 + 0.11x_8 + 0.19x_9 + 0.48x_{10} - 0.56x_{11} + 0.32x_{12}$ OR $\hat{y}=0.20 + 0.16x_1 + 0.45x_3 - 0.01x_4 - 0.55x_5 + 0.39x_6 - 0.18x_7 + 0.20x_8 - 0.11x_9 + 0.82x_{10} - 0.75x_{11} + 0.45x_{12}$ OR $\hat{y}=0.64 - 0.11x_1 + 0.35x_2 + 0.29x_4 - 0.32x_5 + 0.25x_6 - 0.31x_7 + 0.22x_8 + 0.40x_9 + 1.40x_{10} - 1.76x_{11} - 0.13x_{12}$ OR $\hat{y}=0.02 + 0.27x_1 - 0.32x_2 + 0.64x_3 - 0.27x_5 + 0.25x_6 - 0.28x_7 + 0.18x_8 - 0.28x_9 + 0.72x_{10} - 0.30x_{11} - 0.45x_{12}$ OR $\hat{y}=0.09 + 0.21x_1 - 0.53x_2 + 0.64x_3 + 0.26x_4 - 0.01x_6 - 0.39x_7 + 0.15x_8 - 0.14x_9 + 0.72x_{10} - 0.28x_{11} + 0.24x_{12}$ OR $\hat{y}=0.16 + 0.15x_1 - 0.51x_2 + 0.65x_3 + 0.35x_4 - 0.23x_5 - 0.30x_7 + 0.13x_8 - 0.05x_9 + 0.56x_{10} - 0.31x_{11} + 0.35x_{12}$ OR $\hat{y}=0.23 + 0.09x_1 - 0.21x_2 + 0.65x_3 + 0.24x_4 - 1.04x_5 + 0.30x_6 + 0.14x_8 - 0.03x_9 + 0.20x_{10} - 0.39x_{11} + 0.82x_{12}$ OR $\hat{y}=0.38 - 0.09x_1 - 1.06x_2 + 0.81x_3 + 1.13x_4 - 0.39x_5 - 0.55x_6 - 0.25x_7 + 0.35x_9 - 0.11x_{10} + 0.01x_{11} + 0.38x_{12}$ OR $\hat{y}=0.21 + 0.11x_1 - 0.45x_2 + 0.61x_3 + 0.37x_4 - 0.31x_5 + 0.01x_6 - 0.27x_7 + 0.13x_8 + 0.56x_{10} - 0.40x_{11} + 0.35x_{12}$ OR $\hat{y}=0.18 + 0.09x_1 - 0.68x_2 + 0.82x_3 + 0.55x_4 - 0.70x_5 - 0.05x_6 - 0.13x_7 + 0.07x_8 + 0.01x_9 + 0.01x_{11} + 0.70x_{12}$ OR $\hat{y}=0.01 + 0.25x_1 - 0.63x_2 + 0.78x_3 + 0.24x_4 - 0.22x_5 + 0.02x_6 - 0.29x_7 + 0.13x_8 - 0.23x_9 + 0.44x_{10} + 0.49x_{12}$ OR $\hat{y}=0.22 + 0.12x_1 - 0.51x_2 + 0.52x_3 + 0.41x_4 + 0.18x_5 - 0.13x_6 - 0.47x_7 + 0.14x_8 + 0.03x_9 + 0.91x_{10} - 0.55x_{11}$).

In case an ordered set of labels, such as "very cold," "cold," "warm," "hot," and "very hot," is given to the antecedent and consequent trapeziums of each attribute, then the model comes with a level of interpretability as additional asset.

Results for the Capital bike sharing dataset are shown in Table III. The eFGP initial parameters are $\rho = \sigma = 0.30$ and $h_r = 50$. A behavior similar to that observed in Table II for the error indices is noticed in Table III. With the increase of the amount of missing data, the error indices increase monotonically. The number of fuzzy rules tends to increase.

Fig. 5(a) depicts the numerical estimates of the total count of hired bikes for the hardest, 30% MCAR, case. A seasonal pattern

TABLE III
eFGP RESULTS FOR THE CAPITAL BIKE SHARING DATASET ASSUMING MISSING DATA OF THE MCAR TYPE

MCAR	RMSE	NDE	Mean # of rules
0%	0.1090 +/- 0.0015	0.4895 +/- 0.0068	10.4 +/- 0.1
1%	0.1153 +/- 0.0042	0.5178 +/- 0.0186	10.5 +/- 0.8
5%	0.1302 +/- 0.0039	0.5845 +/- 0.0178	8.9 +/- 0.2
10%	0.1403 +/- 0.0057	0.6303 +/- 0.0258	10.5 +/- 0.8
15%	0.1497 +/- 0.0102	0.6724 +/- 0.0459	12.1 +/- 0.4
20%	0.1559 +/- 0.0093	0.7002 +/- 0.0418	13.3 +/- 1.1
30%	0.1597 +/- 0.0098	0.7172 +/- 0.0444	21.8 +/- 2.2

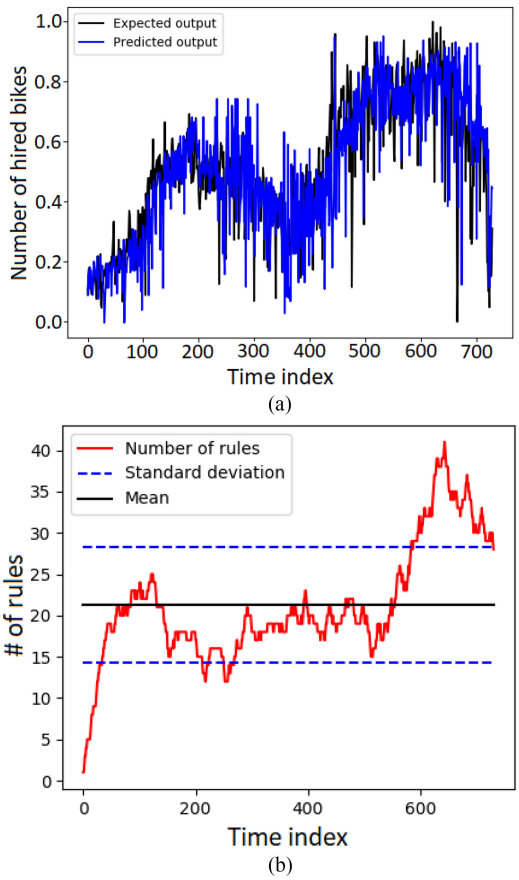

Fig. 5. (a) Daily prediction of the number of hired bikes of the Capital sharing system in Washington, DC, USA, considering eFGP and 30% of MCAR data. (b) Evolution of the number of eFGP rules over time.

(two cycles) is noticed. The number of shared bikes tends to reduce during the winters, as can be seen in the beginning of the time series and after 365 days. The variable, but increasing, trend over the two years is related to the popularization of the loan service. This time series distinguishes from the previously analyzed weather time series mainly because its cycles are longer, i.e., the system dynamics is slower at that time granularity. From one side, a higher number of data samples per cycle can facilitate incremental learning. From the other side, only two cycles with seemingly different aspects are available. Linear and nonadaptive models would certainly have their performance reduced over time due to the changes.

An accurate tracking of the original data can be observed in Fig. 5(a) by using the eFGP pointwise $\hat{y}$. Learning and online

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

development of rules were key points to keep a reasonable prediction accuracy over time. Fig. 5(b) shows the evolution of the eFGP rule base. The granular prediction of the time series, given by the support of active trapezoidal membership functions, is essentially similar to that shown in Fig. 4. The bounds give a range of values around the pointwise prediction, which may be interpreted as a pessimistic and optimistic amount of bike loans in the next day. This may help decision making regarding transfer bikes to different service points, establish new service points, and schedule maintenance.

An example of active rule at $h = 731$ is

R^i: IF (x_1 is $[0.66, 0.66, 0.66, 0.66]$) AND
(x_2 is $[0.75, 0.75, 0.75, 0.75]$) AND
(x_3 is $[0.00, 0.00, 0.00, 0.00]$) AND
(x_4 is $[0.43, 0.57, 0.57, 0.71]$) AND
(x_5 is $[0.00, 0.00, 0.00, 0.00]$) AND
(x_6 is $[0.55, 0.62, 0.74, 0.77]$) AND
(x_7 is $[0.53, 0.57, 0.75, 0.77]$) AND
(x_8 is $[0.42, 0.51, 0.61, 0.64]$) AND
(x_9 is $[0.61, 0.64, 0.79, 0.86]$)

which means if season (x_1) is fall, month (x_2) is September, holiday (x_3) is false, weekdays (x_4) from Tuesday to Thursday, weather (x_5) is clear, temperature (x_6) and apparent temperature (x_7) are high, air humidity (x_8) is moderate, and wind speed (x_9) is high

THEN (y is $[0.63, 0.85, 0.85, 0.91]$) AND

$$(\hat{y} = 0.05 + 0.44x_1 + 0.11x_2 + 0.00x_3 + 0.67x_4 + 0.24x_5 + 0.45x_6 + 0.49x_7 - 1.17x_8 + 0.02x_9 \text{ OR}$$
$$\hat{y} = 0.39 + 0.35x_2 + 0.00x_3 + 0.11x_4 + 1.04x_5 + 0.57x_6 + 0.49x_7 - 1.40x_8 + 0.68x_9 \text{ OR}$$
$$\hat{y} = 0.13 + 0.45x_1 + 0.00x_3 + 0.72x_4 + 0.20x_5 + 0.43x_6 + 0.47x_7 - 1.17x_8 + 0.04x_9 \text{ OR}$$
$$\hat{y} = 0.05 + 0.44x_1 + 0.11x_2 + 0.67x_4 + 0.24x_5 + 0.45x_6 + 0.49x_7 - 1.17x_8 + 0.02x_9 \text{ OR}$$
$$\hat{y} = 0.29 + 0.11x_1 + 0.67x_2 + 0.00x_3 + 1.03x_5 + 0.61x_6 + 0.55x_7 - 1.37x_8 - 0.08x_9 \text{ OR}$$
$$\hat{y} = -0.07 + 0.55x_1 + 0.01x_2 - 0.00x_3 + 0.85x_4 + 0.46x_6 + 0.46x_7 - 1.10x_8 - 0.02x_9 \text{ OR}$$
$$\hat{y} = 0.12 + 0.46x_1 + 0.02x_2 - 0.00x_3 + 0.72x_4 + 0.24x_5 + 0.85x_7 - 1.16x_8 + 0.02x_9 \text{ OR}$$
$$\hat{y} = 0.17 + 0.44x_1 - 0.03x_2 + 0.00x_3 + 0.70x_4 + 0.17x_5 + 0.94x_6 - 1.19x_8 + 0.03x_9 \text{ OR}$$
$$\hat{y} = -3.28 + 1.91x_1 + 0.44x_2 - 0.00x_3 + 2.83x_4 - 2.94x_5 + 0.17x_6 + 1.10x_7 - 0.50x_9 \text{ OR}$$
$$\hat{y} = -0.41 + 0.58x_1 + 0.23x_2 - 0.00x_3 + 0.87x_4 - 0.15x_5 + 1.00x_6 + 0.10x_7 - 1.04x_8).$$

The consequent part indicates that the number of hired bikes will be high, and that the numerical prediction is provided by one of the ten equations depending on whether the input sample is complete or incomplete.

Results for the airfoil self-noise dataset and MCAR data are summarized in Table IV. The eFGP parameters are $\rho = \sigma = 0.1$ and $h_r = 48$. As the percentage of MCAR data increases, a monotonic increase in the number of fuzzy rules and error values is seen. The average RMSE and NDE indices are higher in this problem not only because the standard deviation of the data is higher (as revealed by the greater NDE/RMSE value), but also due to a lower correlation between the attributes and the predicted sound pressure, and to a relatively faster and irregular dynamical behavior. A stochastic component is inherited to the data, and this is reflected on the actual and predicted pressures, as shown in Fig. 6(a) by the rapid amplitude variations. In spite of the noise, we notice that the eFGP approximation follows the trend of the data and captures the changes in the standard deviation.

Fig. 6(b) shows the evolution of the number of eFGP rules for a typical run of the learning algorithm. Initially, a number of rules are created to accommodate never-before-seen data. After about 50 iterations, the model structure becomes more stable, and parameter adaptation prevails over rule creation.

TABLE IV
SUMMARY OF eFGP RESULTS FOR THE AIRFOIL SELF-NOISE DATASET AND MCAR VALUES

MCAR	RMSE	NDE	Mean # of rules
0%	0.1114 +/- 0.0003	0.6059 +/- 0.0017	3.4 +/- 0.1
1%	0.1174 +/- 0.0085	0.6381 +/- 0.0465	5.5 +/- 0.4
5%	0.1251 +/- 0.0056	0.6801 +/- 0.0305	11.5 +/- 1.4
10%	0.1436 +/- 0.0046	0.7808 +/- 0.0249	12.2 +/- 0.9
15%	0.1476 +/- 0.0062	0.8022 +/- 0.0034	12.9 +/- 0.9
20%	0.1502 +/- 0.0073	0.8164 +/- 0.0397	20.4 +/- 1.6
30%	0.1674 +/- 0.0064	0.9102 +/- 0.0351	20.5 +/- 0.6

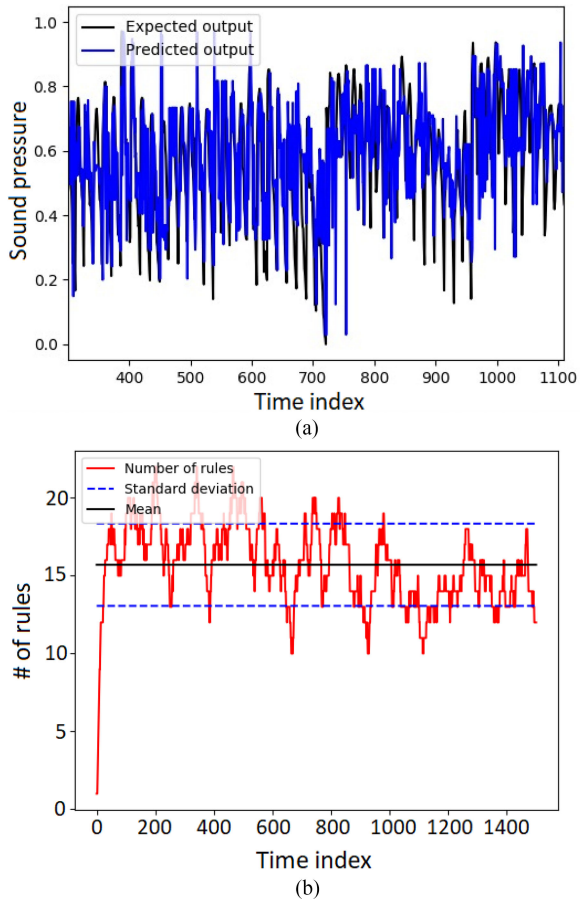

Fig. 6. eFGP. (a) Numerical approximation of the sound pressure for the Airfoil problem. (b) Evolution of the number of rules over time considering 30% of MCAR data.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

TABLE V
eFGP RESULTS FOR THE DEATH VALLEY WEATHER STATION, HIRED BIKES, AND AIRFOIL SELF-NOISE CONSIDERING MAR DATA

Death Valley monthly weather			
MAR	RMSE	NDE	Mean # of rules
5% – 1%	0.0604 +/- 0.0021	0.2335 +/- 0.0082	14.5 +/- 0.9
10% – 1%	0.0611 +/- 0.0010	0.2364 +/- 0.0041	14.4 +/- 0.4
10% – 5%	0.0637 +/- 0.0031	0.2464 +/- 0.0122	20.4 +/- 0.5
20% – 5%	0.0663 +/- 0.0029	0.2566 +/- 0.0112	19.7 +/- 1.6
30% – 5%	0.0632 +/- 0.0015	0.2442 +/- 0.0059	20.7 +/- 0.8
20% – 10%	0.0651 +/- 0.0034	0.2516 +/- 0.0132	29.9 +/- 1.2
30% – 10%	0.0678 +/- 0.0019	0.2622 +/- 0.0075	22.7 +/- 1.2
Number of hired bikes in Washington D.C.			
MAR	RMSE	NDE	Mean # of rules
5% – 1%	0.1216 +/- 0.0066	0.5463 +/- 0.0298	8.0 +/- 0.4
10% – 1%	0.1191 +/- 0.0049	0.5436 +/- 0.0224	7.8 +/- 0.3
10% – 5%	0.1336 +/- 0.0064	0.6001 +/- 0.0287	9.4 +/- 0.6
20% – 5%	0.1262 +/- 0.0048	0.5667 +/- 0.0215	8.9 +/- 0.7
30% – 5%	0.1360 +/- 0.0052	0.6106 +/- 0.0234	8.9 +/- 0.3
20% – 10%	0.1362 +/- 0.0056	0.6116 +/- 0.0253	10.6 +/- 0.9
30% – 10%	0.1383 +/- 0.0041	0.6210 +/- 0.0186	10.3 +/- 0.5
Airfoil sound pressure			
MAR	RMSE	NDE	Mean # of rules
5% – 1%	0.1246 +/- 0.0127	0.6776 +/- 0.0694	4.1 +/- 0.1
10% – 1%	0.1200 +/- 0.0046	0.6523 +/- 0.0254	3.9 +/- 0.1
10% – 5%	0.1210 +/- 0.0060	0.0658 +/- 0.0329	6.8 +/- 0.6
20% – 5%	0.1242 +/- 0.0056	0.6752 +/- 0.0306	11.1 +/- 1.1
30% – 5%	0.1205 +/- 0.0022	0.6552 +/- 0.0120	11.3 +/- 0.7
20% – 10%	0.1315 +/- 0.0040	0.7148 +/- 0.0219	12.4 +/- 1.0
30% – 10%	0.1344 +/- 0.0032	0.7309 +/- 0.0175	11.6 +/- 0.8

The average number of rules is 15.7 on the simulation shown in Fig. 6(b). An interesting event to be observed happens at iteration 720. A sudden reduction of the standard deviation of the data required a new incremental growth of the rule base. In other words, about six rules were added to the eFGP model for an appropriate handling of the concept shift.

An example of rule at $h = 1502$ is

R^i: IF (x_1 is $[0.00, 0.02, 0.10, 0.10]$) AND
(x_2 is $[0.12, 0.15, 0.15, 0.21]$) AND
(x_3 is $[0.25, 0.27, 0.27, 0.29]$) AND
(x_4 is $[0.00, 0.09, 0.10, 0.10]$) AND
(x_5 is $[0.40, 0.43, 0.46, 0.50]$)
THEN (y is $[0.55, 0.56, 0.59, 0.65]$) AND
($\hat{y} = 0.53 - 1.76x_1 - 0.49x_2 + 0.14x_3 + 0.75x_4 + 0.14x_5$
OR
$\hat{y} = 0.24 - 0.31x_2 + 0.06x_3 + 0.75x_4 + 0.33x_5$ OR
$\hat{y} = 0.14 - 1.24x_1 + 0.04x_3 + 0.73x_4 + 0.20x_5$ OR
$\hat{y} = 0.57 - 1.76x_1 - 0.49x_2 + 0.75x_4 + 0.14x_5$ OR
$\hat{y} = 0.96 - 1.76x_1 - 0.39x_2 + 0.26x_3 - 0.39x_5$ OR
$\hat{y} = 0.69 - 2.07x_1 - 0.67x_2 + 0.19x_3 + 0.66x_4$).

The rule can be read as: if frequency (x_1) is very low, angle of attack (x_2) is small, chord length (x_3) is small, free-stream velocity (x_4) is very low, and suction side displacement thickness (x_5) is medium-small, then sound pressure (y) is medium-high.

C. eFGP Results for MAR data

Experiments with MAR data were conducted for the same datasets. Table V summarizes the results. In general, the RMSE and NDE indices vary irregularly with the increase of MAR values—although a modest performance reduction is observed from the extreme cases. The size of the rule base increases with the amount of MAR data in most cases. However, the increase of the model structure is not in the same proportion as that observed for MCAR data. MCAR data impose more challenges to incremental modeling. In other words, if readings from a single attribute become partially available for some reason, the eFGP learning algorithm still benefits from the information of the other attributes to keep its prediction accuracy. Contrariwise, if the availability of readings of all attributes is limited, then the algorithm uses approximations from nearest granules to provide predictions.

The result for Death Valley, shown in Table V, portrays that the maximum difference in average RMSE between the easiest (5%–1%) and the roughest (30%–10%) MAR cases is 0.0074 – a 10.9% increase. For the bike sharing and airfoil sound pressure problems, the same difference is of 13.9% and 10.7%, respectively. While 30% of missing values can be quite detrimental to other prediction models (as will be empirically shown in the next section), the eFGP copes with MAR values in an evolving fashion, thus keeping reasonable RMSE and NDE rates. Notice also in Table V that the standard deviation of the error indices for ten runs of the algorithm for each MAR case is minimal. The eFGP learning approach has provided prediction models that are robust to MAR values.

In summary, the eFGP has shown to be able to handle non-stationary data streams containing MCAR and MAR values at different rates. The behavior of the algorithm has been stable in different real missing-data scenarios. MAR data are more easily dealt with by the algorithm compared to MCAR data. The latter requires a greater number of information granules and a larger expansion of the fuzzy rule base, whereas parametric adaptation prevails in the former case.

D. Comparing Evolving Intelligent Models

eFGP models are compared with eGNN [45], eTS [46], xTS [47], and FBeM [42]. In the Death Valley problem, the eFGP uses $\rho = \sigma = 0.3$ and $h_r = 150$. The eGNN uses min–max neurons, $\rho^{[0]} = 0.2, h_r = 80, \zeta = 0.9$ (a constant for updating connection weights), and $\eta = 0.5$ (a threshold for the number of neurons created in h_r time steps). eTS uses $\Omega = 350$ and radius length, $r = 0.7$; xTS employs $\Omega = 350$; and FBeM uses $\rho = \sigma = 0.3$ and $h_r = 150$. These parameters provided the highest average accuracy of each method based on ten runs. As other methods are not supplied with mechanisms to impute missing data, a zero-order-hold approach is employed, i.e., the last prediction is replicated.

Fig. 7 depicts the RMSE indices of the predictors for Death Valley using different proportions of missing values. Clearly, the increasing rate of the eFGP is less than those of the other models. The eFGP takes full advantage of the information within incomplete samples. The correlation matrix between attributes becomes more distorted if entire samples are removed from the stream. This more strongly hampers the ability of the other models to give accurate predictions. Fig. 7(b) highlights that the percentage of MAR values related to the attributes with less missing values is more influential to eTS, xTS, eGNN, and FBeM. The eFGP approach has demonstrated robustness to MAR data. Table VI gives a summary of the results achieved for all methods and situations illustrated in Fig. 7.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

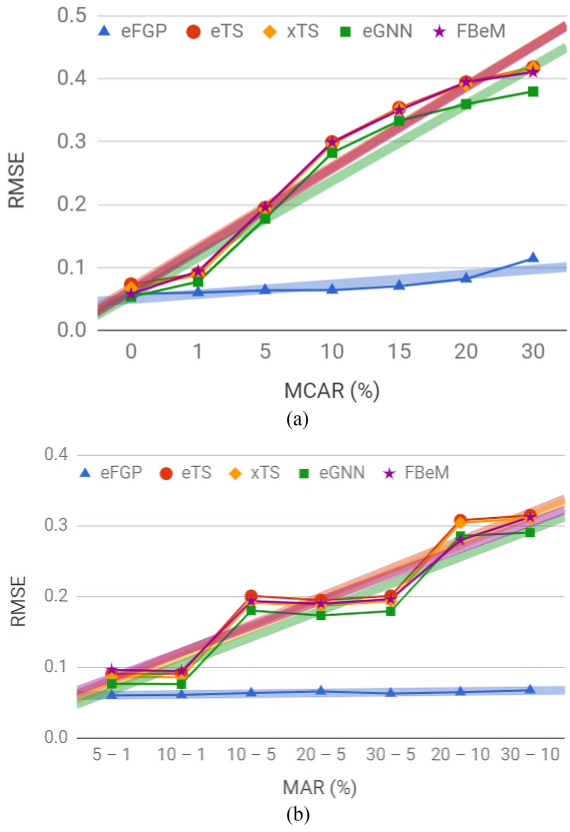

Fig. 7. Performance comparison on the prediction of the Death Valley monthly temperature. (a) MCAR data. (b) MAR data.

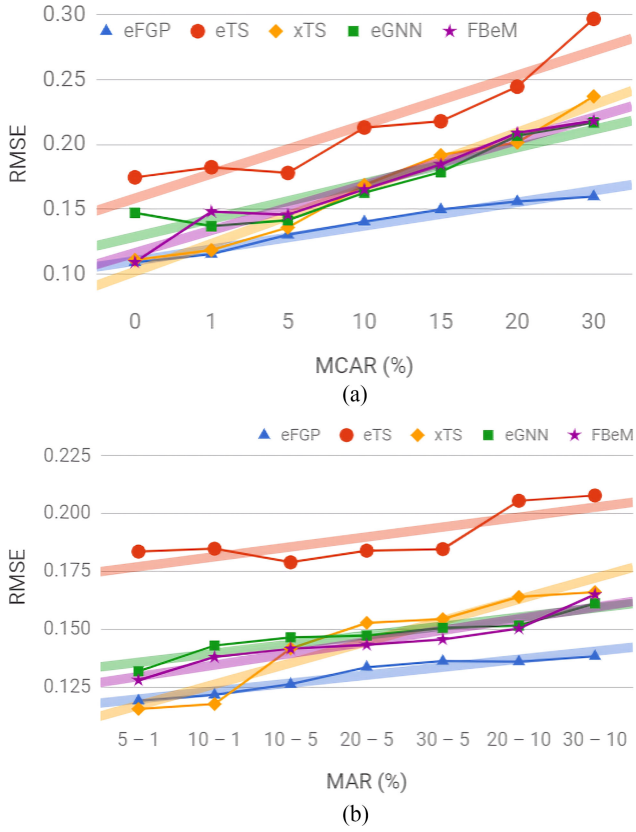

Fig. 8. Performance comparison on the prediction of the Bike sharing dataset. (a) MCAR data. (b) MAR data.

TABLE VI
SUMMARY RESULTS FOR DEATH VALLEY

Complete dataset			
Method	RMSE	NDE	# of Rules
eFGP	0.0579 +/- 0.0018	0.2239 +/- 0.0070	13.3 +/- 0.1
eGNN	**0.0532 +/- 0.0019**	**0.2055 +/- 0.0071**	14.0 +/- 0.8
eTS	0.0730 +/- 0.0000	0.2822 +/- 0.0000	11.0 +/- 0.0
xTS	0.0669 +/- 0.0000	0.2586 +/- 0.0000	**8.0 +/- 0.0**
FBeM	0.0579 +/- 0.0018	0.2239 +/- 0.0070	13.3 +/- 0.1
Average results for MCAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.1142 +/- 0.0057**	**0.4180 +/- 0.0222**	28.1 +/- 2.3
eGNN	0.3796 +/- 0.0237	1.4673 +/- 0.0917	8.9 +/- 0.7
eTS	0.4180 +/- 0.0095	1.6153 +/- 0.0368	6.4 +/- 0.9
xTS	0.4178 +/- 0.0432	1.6149 +/- 0.1670	**3.8 +/- 1.5**
FBeM	0.4107 +/- 0.0182	1.5873 +/- 0.0702	9.8 +/- 0.9
Average results for MAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.0678 +/- 0.0019**	**0.2622 +/- 0.0075**	22.7 +/- 1.2
eGNN	0.2906 +/- 0.0088	1.1233 +/- 0.0340	14.8 +/- 0.7
eTS	0.3150 +/- 0.0160	1.2174 +/- 0.0618	8.6 +/- 1.1
xTS	0.3106 +/- 0.0138	1.2004 +/- 0.0536	**5.4 +/- 1.1**
FBeM	0.3126 +/- 0.0168	1.2081 +/- 0.0651	12.9 +/- 0.4

TABLE VII
SUMMARY RESULTS FOR THE BIKE LOANS

Complete dataset			
Method	RMSE	NDE	# of Rules
eFGP	0.1090 +/- 0.0015	0.4895 +/- 0.0068	**10.4 +/- 0.1**
eGNN	0.1471 +/- 0.0007	0.6602 +/- 0.0032	11.0 +/- 0.1
eTS	0.1745 +/- 0.0000	0.7829 +/- 0.0000	16.0 +/- 0.0
xTS	**0.1078 +/- 0.0000**	**0.4854 +/- 0.0000**	16.0 +/- 0.0
FBeM	0.1090 +/- 0.0015	0.4895 +/- 0.0068	**10.4 +/- 0.1**
Average results for MCAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.1597 +/- 0.0098**	**0.7172 +/- 0.0444**	21.8 +/- 2.2
eGNN	0.2168 +/- 0.0153	0.9720 +/- 0.0665	11.7 +/- 1.3
eTS	0.2967 +/- 0.0621	1.3312 +/- 0.2788	10.8 +/- 2.9
xTS	0.2370 +/- 0.0218	1.0630 +/- 0.0980	**6.4 +/- 1.8**
FBeM	0.2180 +/- 0.0170	0.9816 +/- 0.0664	11.6 +/- 1.2
Average results for MAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.1383 +/- 0.0041**	**0.6210 +/- 0.0186**	**10.3 +/- 0.5**
eGNN	0.1515 +/- 0.0103	0.6801 +/- 0.0464	11.7 +/- 1.3
eTS	0.2055 +/- 0.0477	0.9219 +/- 0.2140	16.0 +/- 4.9
xTS	0.1661 +/- 0.0221	0.7451 +/- 0.0989	12.0 +/- 1.6
FBeM	0.1650 +/- 0.0121	0.7400 +/- 0.0423	11.7 +/- 1.4

For the bike sharing dataset, eFGP models using $\rho = \sigma = 0.5$ and $h_r = 50$ were compared to: eGNN using min–max neurons, $\rho^{[0]} = 0.5$, $h_r = 50$, $\zeta = 0.9$, and $\eta = 1.5$; eTS with $\Omega = 350$ and $r = 0.3$; xTS using $\Omega = 350$; and FBeM with $\rho = \sigma = 0.5$ and $h_r = 50$. Fig. 8(a) shows that the eFGP is more robust than the other methods for MCAR data across the range of analyzed values. xTS presented the lowest RMSE for the complete dataset. However, removing a small percentage of random values from the data stream is enough for eFGP to overcome xTS. MAR results are shown in Fig. 8(b). Notably, comparing the extreme 5–1 and 30–10 cases, eFGP is the model with the smallest growth rate. This argues in favor of its greater robustness. The percentage of missing values related to the attributes with less missing values is more influential to the predictions. Table VII summarizes the average accuracy and number of rules of the predictors. xTS and eFGP are the most accurate predictors of the

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

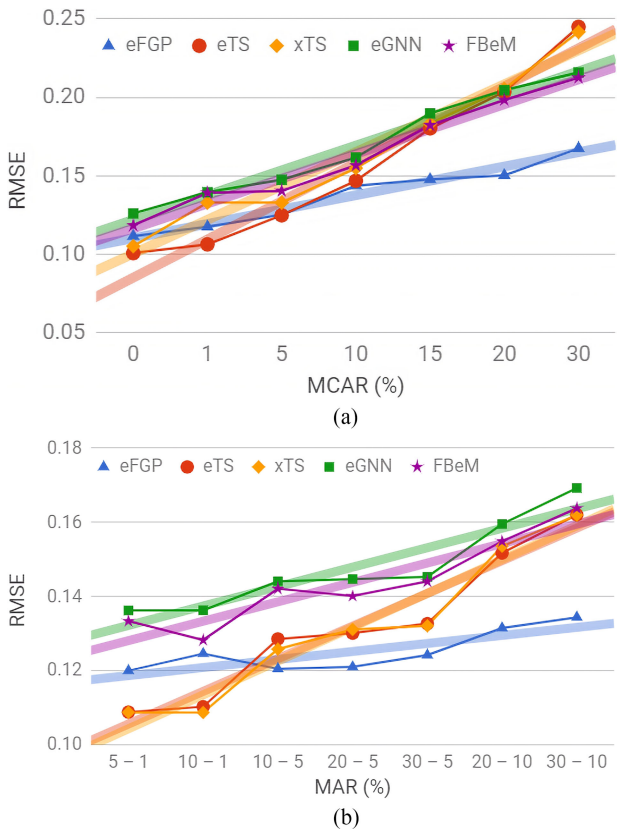

Fig. 9. Performance comparison on the prediction of the Airfoil self-noise dataset. (a) MCAR data. (b) MAR data.

amount of loans when the dataset is complete and incomplete, respectively.

In the airfoil sound pressure problem, the eFGP uses $\rho = \sigma = 0.6$ and $h_r = 48$; eGNN employs min–max neurons, $\rho^{[0]} = 0.2$, $h_r = 80$, $\zeta = 0.9$, and $\eta = 1.5$; eTS uses $\Omega = 100$ and $r = 0.9$; xTS utilizes $\Omega = 100$; and FBeM uses $\rho = \sigma = 0.6$ and $h_r = 48$. Fig. 9 shows the average performance of the models for MCAR and MAR cases. An increasing trend is noticed for all models. The eFGP has been the most robust predictor in all scenarios. Although eTS and xTS are more accurate for the complete dataset, when about 3% of the data, in general, are missing, the eFGP begins to prevail.

Table VIII compares error indices and number of rules. Even though the eFGP uses only 3.4 granules against nine and 12 clusters of eTS and xTS for the complete data, its error indices are close to those of eTS and xTS. Malfunction of sensors or failures in data communication can quickly deteriorate the performance of evolving models, as shown in the table for the missing-data cases. The eFGP overcame eGNN, eTS, xTS, and FBeM by 23.4%, 31.6%, 30.7%, and 21.2% in MCAR scenarios, and by 20.6%, 17.0%, 17.0%, and 17.9% in MAR settings, respectively.

The average time taken by the eFGP to process a sample on an Intel i7 3.6-GHz processor with 16-GB RAM using Python-Ubuntu 18.04 on Windows 10 was 20.5, 11.6, and 9.3 ms, respectively, for the Death Valley, Bike Sharing, and Self-Noise datasets. FBeM spent 17.8, 10.7, and 8.4 ms; eTS spent 24.2, 15.5, and 9.9 ms; xTS spent 17.2, 10.9, and 8.6 ms; and eGNN consumed 23.9, 16.5, and 11.0 ms. The eFGP is competitive with the other methods in processing time.

TABLE VIII
SUMMARY RESULTS FOR THE AIRFOIL SOUND PRESSURE

Complete dataset			
Method	RMSE	NDE	# of Rules
eFGP	0.1114 +/- 0.0003	0.6059 +/- 0.0017	**3.4 +/- 0.1**
eGNN	0.1259 +/- 0.0003	0.6839 +/- 0.0019	14.9 +/- 0.6
eTS	**0.1006 +/- 0.0000**	**0.5467 +/- 0.0000**	9.0 +/- 0.0
xTS	0.1050 +/- 0.0000	0.5705 +/- 0.0000	12.0 +/- 0.0
FBeM	0.1114 +/- 0.0003	0.6059 +/- 0.0017	**3.4 +/- 0.1**
Average results for MCAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.1674 +/- 0.0064**	**0.9102 +/- 0.0351**	20.2 +/- 0.6
eGNN	0.2158 +/- 0.0058	1.1725 +/- 0.0316	45.5 +/- 1.4
eTS	0.2449 +/- 0.0183	1.3307 +/- 0.0994	12.0 +/- 5.3
xTS	0.2416 +/- 0.0199	1.3130 +/- 0.1079	**9.2 +/- 1.9**
FBeM	0.2125 +/- 0.0141	1.1673 +/- 0.0329	45.5 +/- 1.4
Average results for MAR scenarios			
Method	RMSE	NDE	# of Rules
eFGP	**0.1344 +/- 0.0032**	**0.7309 +/- 0.0175**	11.6 +/- 0.8
eGNN	0.1693 +/- 0.0060	0.9197 +/- 0.0324	26.0 +/- 2.2
eTS	0.1620 +/- 0.0106	0.8804 +/- 0.0575	**9.4 +/- 0.6**
xTS	0.1620 +/- 0.0068	0.8801 +/- 0.0367	9.4 +/- 2.3
FBeM	0.1638 +/- 0.0076	0.9129 +/- 0.0336	26.0 +/- 2.3

E. Statistical Hypothesis Testing

Balanced one-way analysis of variance (ANOVA) [48] compares evolving models under the same roof, regardless of the application. ANOVA determines if the mean accuracies of a pair of methods are statistically different. The null hypothesis is that the mean accuracy of the methods is essentially the same. A cutoff value, p, less than 0.05 indicates that the accuracy of at least one of the methods is significantly different from the others. Considering the results of all experiments with complete datasets, a $p = 0.7372$ was obtained, and therefore, the null hypothesis holds true. In other words, any method, eFGP, eGNN, eTS, xTS, and FBeM, may provide the best estimates for a particular stream if no data are missing. The eFGP is competitive with state-of-the-art evolving methods.

From the results of the MCAR and MAR experiments, the values $p = 0.0015$ and $p = 0.0031$, respectively, were obtained. The mean accuracy of the methods is not all the same, i.e., the null hypothesis is rejected. The Tukey honestly significant difference test [48] was performed to compare pairs of methods. The Tukey test is optimal for balanced one-way ANOVA. Table IX shows the results of the test for a 95% confidence interval (CI) for the true difference of the means.

A negative or positive "difference of means" denotes that the first or second method, respectively, outperformed the other on that scenario. However, if the CI [LB, UB] (see Table IX) contains the value 0, then the difference between the methods is not significant at the 0.05 level. In this case, for applications other than those evaluated, any of the methods can be better than the other, and both should ideally be examined. Notice from the MAR and MCAR scenarios in Table IX that the eFGP is statistically superior. eGNN, eTS, xTS, and FBeM are similar among themselves in missing-data scenarios.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

TABLE IX
TUKEY TEST RESULTS

Tukey Test results with complete dataset				
Method 1	Method 2	CI LB	Diff of Means	CI UB
eFGP	eGNN	-0.2090	-0.0464	0.1163
eFGP	eTS	-0.2230	-0.0604	0.1023
eFGP	xTS	-0.1621	0.0006	0.1632
eFGP	FBeM	-0.1626	0	0.1626
eGNN	eTS	-0.1767	-0.0140	0.1486
eGNN	xTS	-0.1157	0.0469	0.2096
eGNN	FBeM	-0.1163	0.0464	0.2090
eTS	xTS	-0.1017	0.0610	0.2236
eTS	FBeM	-0.1023	0.0604	0.2230
xTS	FBeM	-0.1632	-0.0006	0.1621
Tukey Test results for MCAR scenarios				
Method 1	Method 2	CI LB	Diff of Means	CI UB
eFGP	eGNN	-0.6443	-0.3229	-0.0014
eFGP	eTS	-0.7798	-0.4584	-0.1369
eFGP	xTS	-0.7216	-0.4001	-0.0786
eFGP	FBeM	-0.6699	-0.3485	-0.0270
eGNN	eTS	-0.4569	-0.1355	0.1860
eGNN	xTS	-0.3987	-0.0772	0.2442
eGNN	FBeM	-0.3470	-0.0256	0.2959
eTS	xTS	-0.2632	0.0583	0.3797
eTS	FBeM	-0.2116	0.1099	0.4314
xTS	FBeM	-0.2698	0.0516	0.3731
Tukey Test results for MAR scenarios				
Method 1	Method 2	CI LB	Diff of Means	CI UB
eFGP	eGNN	-0.4529	-0.2300	-0.0071
eFGP	eTS	-0.5141	-0.2913	-0.0684
eFGP	xTS	-0.4745	-0.2516	-0.0287
eFGP	FBeM	-0.4808	-0.2580	-0.0351
eGNN	eTS	-0.2842	-0.0613	0.1616
eGNN	xTS	-0.2445	-0.0216	0.2012
eGNN	FBeM	-0.2509	-0.0280	0.1949
eTS	xTS	-0.1832	0.0396	0.2625
eTS	FBeM	-0.1896	0.0333	0.2562
xTS	FBeM	-0.2292	-0.0063	0.2165

LB: Lower bound; UB: Upper bound; CI: Confidence interval.

In addition to the preeminence of eFGP in terms of overall accuracy in missing-data context, the granular approximation of the underlying time series or data stream and the linguistic description associated with fuzzy granules are distinctive features to consider the eFGP approach.

V. CONCLUSION

In this contribution, we shed light on the question of missing values in nonstationary data streams. We described an evolving granular fuzzy-rule-based modeling method for function approximation and time-series prediction in online settings, where values may be MAR and MCAR. eFGP models can handle single and multiple missing values per sample due to its constructive features, namely, 1) a modified rule structure that includes reduced-term consequent functions, and conjunctive and disjunctive operators; and 2) a learning algorithm customized to such modified structure, which uses partial similarity, and time-varying granules.

An extensive set of experimental results on actual weather, social service, and engineering applications considering from 1% to 30% of missing values has shown that the eFGP approach outperforms other evolving fuzzy and neuro-fuzzy modeling methods that resort to sample deletion and replication of the last output. Moreover, the results are statistically significant on MAR and MCAR scenarios according to an ANOVA–Tukey test. A particular characteristic of the eFGP concerns the provision of a granular enclosure of the underlying time series or data stream, which may assist decision making and have a variety of interpretations in different areas. The eFGP approach may inspire fundamental modifications of other computational intelligence methods to include the capability of dealing with missing data and providing numerical and granular estimates in a nonlinear and time-varying way, considering the properties and changes of data streams.

Further work will discuss missing-data imputation in semisupervised multiclass classification of data streams. The development of fuzzy granules with different geometries and incremental adaptation of the parameters of aggregation operators will be discussed considering streams of nonstationary data subject to missing values. Imputation of nominal values and imputation in information-retrieval and natural-language-processing contexts will also be discussed.

REFERENCES

[1] D. Leite, "Evolving granular systems," Ph.D. dissertation, School Elect. Comput. Eng., Univ. Campinas, Campinas, Brazil, 2012.
[2] J. Gama, *Knowledge Discovery from Data Streams*. Boca Raton, FL, USA: CRC Press, 2010.
[3] P. Angelov, X. Gu, and J. C. Principe, "Autonomous learning multimodel systems from data streams," *IEEE Trans Fuzzy Syst.*, vol. 26, no. 4, pp. 2213–2224, Aug. 2018.
[4] D. Leite, R. M. Palhares, V. C. Campos, and F. Gomide, "Evolving granular fuzzy model-based control of nonlinear dynamic systems," *IEEE Trans Fuzzy Syst.*, vol. 23, no. 4, pp. 923–938, Aug. 2015.
[5] D. Dovžan, V. Logar, and I. Škrjanc, "Implementation of an evolving fuzzy model (eFuMo) in a monitoring system for a water treatment process," *IEEE Trans Fuzzy Syst.*, vol. 23, no. 5, pp. 1761–1776, Oct. 2015.
[6] Z. Mirzamomen and M. R. Kangavari, "Evolving fuzzy min–max neural network based decision trees for data stream classification," *Neural Process. Lett.*, vol. 45, no. 1, pp. 341–363, 2017.
[7] J. J. Rubio, "USNFIS: Uniform stable neuro fuzzy inference system," *Neurocomputing*, vol. 262, pp. 57–66, 2017.
[8] R. Hyde, P. Angelov, and A. R. MacKenzie, "Fully online clustering of evolving data streams into arbitrarily shaped clusters," *Inf. Sci.*, vol. 382–383, pp. 96–114, 2017.
[9] M. Pratama, J. Lu, E. Lughofer, G. Zhang, and M. J. Er, "An incremental learning of concept drifts using evolving type-2 recurrent fuzzy neural networks," *IEEE Trans Fuzzy Syst.*, vol. 25, no. 5, pp. 1175–1192, Oct. 2017.
[10] E. Lughofer and M. Pratama, "Online active learning in data stream regression using uncertainty sampling based on evolving generalized fuzzy models," *IEEE Trans Fuzzy Syst.*, vol. 26, no. 1, pp. 292–309, Feb. 2018.
[11] G. Andonovski, G. Mušič, S. Blažič, and I. Škrjanc, "Evolving model identification for process monitoring and prediction of non-linear systems," *Eng. Appl. Artif. Intell.*, vol. 68, pp. 214–221, 2018.
[12] E. Soares, P. Costa, B. Costa, and D. Leite, "Ensemble of evolving data clouds and fuzzy models for weather time series prediction," *Appl. Soft Comput.*, vol. 64, pp. 445–453, 2017.
[13] I. Škrjanc, J. Iglesias, A. Sanchis, D. Leite, E. Lughofer, and F. Gomide, "Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: A survey," *Inf. Sci.*, vol. 490, pp. 344–368, 2019.
[14] I. Aydilek and A. Arslan, "A novel hybrid approach to estimating missing values in databases using K-nearest neighbors and neural networks," *Int. J. Innov. Comput., Inf. Control*, vol. 7, no. 8, pp. 4705–4717, 2012.
[15] T. D. Pigott, "A review of methods for missing data," *Educ. Res. Eval.*, vol. 7, no. 4, pp. 353–383, 2001.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.

[16] A. Farhangfar, L. A. Kurgan, and W. Pedrycz, "A novel framework for imputation of missing values in databases," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 37, no. 5, pp. 692–709, Sep. 2007.
[17] J. Luengo, S. García, and F. Herrera, "On the choice of the best imputation methods for missing values considering three groups of classification methods," *Knowl. Inf. Syst.*, vol. 32, pp. 77–108, 2012.
[18] V. Audigier *et al.*, "Multiple imputation for multilevel data with continuous and binary variables," *Statist. Sci.*, vol. 33, no. 2, pp. 160–183, 2018.
[19] O. Troyanskaya *et al.*, "Missing value estimation methods for DNA microarrays," *Bioinformatics*, vol. 17, no. 6, pp. 520–525, 2001.
[20] M. Amiri and R. Jensen, "Missing data imputation using fuzzy-rough methods," *Neurocomputing*, vol. 205, pp. 152–164, 2016.
[21] M. L. Yadav and B. Roychoudhury, "Handling missing values: A study of popular imputation packages in R," *Knowl. Based Syst.*, vol. 160, pp. 104–118, 2018.
[22] J. J. Rubio and A. Bouchachia, "MSAFIS: An evolving fuzzy inference system," *Soft Comput.*, vol. 21, no. 9, pp. 2357–2366, 2017.
[23] P. Angelov, *Autonomous Learning Systems: From Data Streams to Knowledge in Real-Time*. Chichester, U.K.: Wiley, 2012.
[24] W. Pedrycz, A. Skowron, and V. Kreinovich, *Handbook of Granular Computing*. Chichester, U.K.: Wiley, 2008.
[25] R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data* (Wiley Series in Probability and Statistics), 2nd ed. Hoboken, NJ, USA: Wiley, 2002.
[26] F. Lobato *et al.*, "Multi-objective genetic algorithm for missing data imputation," *Pattern Recognit. Lett.*, vol. 68, pp. 126–131, 2015.
[27] C. Hopkin, R. Hoyle, and N. Gottfredson, "Maximizing the yield of small samples in prevention research: A review of general strategies and best practices," *Prevention Sci.*, vol. 16, no. 7, pp. 950–955, 2015.
[28] I. Aydilek and A. Arslan, "A hybrid method for imputation of missing values using optimized fuzzy *c*-means with support vector regression and a genetic algorithm," *Inf. Sci.*, vol. 233, pp. 25–35, Jun. 2013.
[29] S. Azim and S. Aggarwal, "Using fuzzy *c* means and multi layer perceptron for data imputation: Simple v/s complex dataset," in *Proc. IEEE Int. Conf. Recent Adv. Inf. Technol.*, 2016, pp. 197–202.
[30] D. Li, H. Gu, and L. Zhang, "A hybrid genetic algorithm–Fuzzy *c*-means approach for incomplete data clustering based on nearest-neighbor intervals," *Soft Comput.*, vol. 17, no. 10, pp. 1787–1796, 2013.
[31] P. Saravanan and P. Sailakshmi, "Missing value imputation using fuzzy possibilistic *c*-means optimized with support vector regression and genetic algorithm," *J. Theor. Appl. Inf. Technol.*, vol. 72, no. 1, 2015.
[32] V. Kuppusamy and I. Paramasivam, "Grey fuzzy neural network-based hybrid model for missing data imputation in mixed database," *Int. J. Intell. Eng. Syst.*, vol. 10, no. 3, pp. 146–155, 2017.
[33] L. Zhang, W. Lu, X. Liu, W. Pedrycz, C. Zhong, and L. Wang, "A global clustering approach using hybrid optimization for incomplete data based on interval reconstruction of missing value," *Int. J. Intell. Inf. Technol.*, vol. 31, no. 4, pp. 297–313, 2016.
[34] N. Samat and M. Salleh, "A study of data imputation using fuzzy C-means with particle swarm optimization," in *Proc. Int. Conf. Soft Comput. Data Mining*, 2017, pp. 91–100.
[35] M. Bańbura and M. Modugno, "Maximum likelihood estimation of factor models on datasets with arbitrary pattern of missing data," *J. Appl. Econ.*, vol. 29, no. 1, pp. 133–160, 2014.
[36] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning* (Springer Series in Statistics), vol. 1. New York, NY, USA: Springer, 2001.
[37] H. Li, X. Deng, and E. Smith, "Missing data imputation for paired stream and air temperature sensor data," *Environmetrics*, vol. 28, no. 1, 2017, Art. no. e2426.
[38] I. Žliobaitė and J. Hollmén, "Optimizing regression models for data streams with missing values," *Mach. Learn.*, vol. 99, pp. 47–73, 2015.
[39] K. M. Lang and T. D. Little, "Principled missing data treatments," *Prevention Sci.*, vol. 19, no. 3, pp. 284–294, 2018.
[40] J. Graham, "Adding missing-data-relevant variables to FIML-based structural equation models," *Struct. Equ. Model.*, vol. 10, pp. 80–100, 2003.
[41] D. Leite, F. Gomide, R. Ballini, and P. Costa, "Fuzzy granular evolving modeling for time series prediction," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2011, pp. 2794–2801.
[42] D. Leite, R. Ballini, P. Costa, and F. Gomide, "Evolving fuzzy granular modeling from nonstationary fuzzy data streams," *Evol. Syst.*, vol. 3, no. 2, pp. 65–79, 2012.
[43] S. Roof and C. Callagan, "The climate of Death Valley, California," *Bull. Amer. Meteorol. Soc.*, vol. 84, pp. 1725–1739, 2003.
[44] T. F. Brooks, D. S. Pope, and A. M. Marcolini, "Airfoil self-noise and prediction," NASA, Washington, DC, USA, Tech. Rep. RP-1218, 1989.
[45] D. Leite, P. Costa, and F. Gomide, "Evolving granular neural networks from fuzzy data streams," *Neural Netw.*, vol. 38, pp. 1–16, 2013.
[46] P. Angelov and D. Filev, "An approach to online identification of Takagi-Sugeno fuzzy models," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 484–498, Feb. 2004.
[47] P. Angelov and X. Zhou, "Evolving fuzzy systems from data streams in real-time," in *Proc. Int. Symp. Evol. Fuzzy Syst.*, 2006, pp. 29–35.
[48] B. Antonisamy, P. S. Premkumar, and S. Christopher, *Principles and Practice of Biostatistics*. New York, NY, USA: Elsevier, 2017.

Authorized licensed use limited to: Otto-von-Guericke Universitaet Magdeburg. Downloaded on January 07,2024 at 20:45:08 UTC from IEEE Xplore. Restrictions apply.