ORIGINAL PAPER

Variable step-size evolving participatory learning with kernel recursive least squares applied to gas prices forecasting in Brazil

Eduardo Ravaglia Campos Queiroz[1] · Kaike Sa Teles Rocha Alves[2] · Fernando Luiz Cyrino Oliveira[1] · Eduardo Pestana de Aguiar[3]

Abstract

A prediction model is an indispensable tool in business, helping to make decisions, whether in the short, medium, or long term. In this context, the implementation of machine learning techniques in time series forecasting models has a notorious relevance, as information processing and efficient and dynamic knowledge uncovering are increasingly demanded. This paper develops a model called Variable step-size evolving Participatory Learning with Kernel Recursive Least Squares, VS-ePL-KRLS, applied to the forecast of weekly prices for S500 and S10 diesel oil, at the Brazilian level, for biweekly and monthly horizons. The presented model demonstrates a better accuracy compared with analogous models in the literature, without loss of computational performance for all time series analyzed.

Keywords Forecasting · Time series · Evolving fuzzy models · Variable step-size

1 Introduction

A prediction is an essential tool in business, where it helps in making decisions about operational (short term), tactical (medium term), and strategic (long term) planning (Hyndman and Athanasopoulos 2018). In a world that demands information processing and knowledge extraction in an increasingly efficient and dynamic way, the combination of Forecasting and Machine Learning techniques has gained notorious relevance in modeling real-world phenomena, which are characterized by complexity and uncertainties.

✉ Fernando Luiz Cyrino Oliveira
  cyrino@puc-rio.br

  Eduardo Ravaglia Campos Queiroz
  eduardo.campos@engenharia.ufjf.br

  Kaike Sa Teles Rocha Alves
  kaike.alves@engenharia.ufjf.br

  Eduardo Pestana de Aguiar
  eduardo.aguiar@engenharia.ufjf.br

[1] Department of Industrial Engineering, Pontifical Catholic University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil

[2] Graduate Program in Computational Modeling, Federal University of Juiz de Fora, Juiz de Fora, MG, Brazil

[3] Department of Industrial and Mechanical Engineering, Federal University of Juiz de Fora, Juiz de Fora, MG, Brazil

Several authors have developed models to achieve the optimal point at the trade-off between the quality of decisions and response times.

In recent years, a new family of methods emerged in the context of online machine learning and it became proeminent in nonstationary time series prediction. It was stimulated due to the increasing availability of data and the need to real-time predict their behavior to support decision-making. Evolving fuzzy algorithms are an important segment of evolving systems because they allow a model to react to data changes by creating, merging, updating, and deleting fuzzy rules (Kasabov and Filev 2006). A wide variety of fuzzy algorithms and models can be found in the literature. The authors in Lughofer (2016) discussed recent advances that have improved stability, reliability, and useability as well as aspects related to grown-up interpretability for real-world applications such as on-line condition monitoring, visual inspection, human-machine interaction, smart sensors, production systems and others. Another important contribution is presented by Škrjanc et al. (2019), a systematic survey on evolving intelligent systems with a focus on fuzzy and neuro-fuzzy methods for clustering, classification, regression and system identification in real-world applications as online trading, financial analysis, e-commerce and business, smart home, health care, transportation systems, global

supply logistic chains, smart grids, industrial control, cyber-security, and many other areas.

The evolving Takagi–Sugeno (eTS) model is considered the precursor (Angelov and Filev 2004). This model updates the system recursively, adding new rules, or updating existing rules. Antecedent terms of TS rules are determined through a subtractive clustering approach (Chiu 1994) based on the notion of potential function. Consequent parameters of TS rules are updated through Recursive Least Squares (Young 2012). Other models have modified the eTS, such as eTS+ (Angelov et al. 2010), which use concepts such as age, utility, local density, and zone of influence to update a rule base; and eTS-KRLS (Shafieezadeh-Abadeh and Kalhor 2016) whose consequent TS parameters are updated through Kernel Recursive Least Squares (Engel et al. 2004).

Lima et al. (2010) proposed an algorithm, evolving Participatory Learning (ePL), which uses an unsupervised dynamic fuzzy clustering algorithm that mimics human learning (Yager 2004). In this approach, updating the rule base is a function of existing rules and new information received. A similar process occurs with human learning, which is amplified if there exists prior knowledge on a given subject. Maciel et al. (2017) extended the initial ideas with the development of ePL+, which, similar to Angelov et al. (2010), uses criteria such as utility, age, area of influence, and local density to update the rule base. Both models, ePL and ePL+, uses the Recursive Least Squares method (RLS) to estimate consequent parameters.

Recently, (Vieira et al. 2018) proposed a new method called ePL-KRLS, inspired by Shafieezadeh-Abadeh and Kalhor (2016). The main difference between ePL-KRLS and ePL or (ePL+) is the way the consequent parameters are estimated. The former uses the Kernel Recursive Least Squares (KRLS). The advantage of using this approach is that Kernel-based methods are more sensitive to variations in the input data and are able to approximate nonlinear systems accurately and efficiently with moderate computational cost Vieira et al. (2018). Alves et al. (2020) introduced the methods called SM-ePL-KRLS and ESM-ePL-KRLS. These models work with the recursive adjustment of parameters according to the error based on the Set-Membership (SM) de Aguiar et al. (2017) and Enhanced Set-Membership (ESM) Alves et al. (2020) filtering.

The present paper proposes a forecasting model called Variable Step-Size evolving Participatory Learning with Kernel Recursive Least Squares (VS-ePL-KRLS). It expands the ePL-KRLS model with an error-based variable-step-size (VS) parameter-updating method. The VS algorithm, discussed in Harris et al. (1986), is an extension of earlier ideas in stochastic approximation using changeable the step size in gradient methods. According to Evans et al. (1993), Variable Step-Size methods are an attempt to improve the convergence of the Least Mean Squares (LMS) algorithm

while preserving the steady-state performance; the increase in the complexity of the implementation is relatively low. Compared to the SM-ePL-KRLS and ESM-ePL-KRLS models, it is easier to implement and, consequently, further computational complexity savings are expected as noted by de Aguiar et al. (2017).

The main contributions of this work are summarized as follows:

– We introduce a new forecasting model based on the Variable Step-Size (Harris et al. 1986) to adjust the parameter that controls the rate of change of the arousal index.
– We approach an unexplored and important class of time series, which is the price of fuels, more specifically Brazilian S500 and S10 diesel oil, with weekly data periodicity for biweekly and monthly horizons.
– We evaluate the performance of the proposed model in terms of the average number of rules, error metrics, and percentage of system-wide CPU utilization. Additionally, we compare the performance with benchmarks evolving models like ePL-KRLS (Vieira et al. 2018), SM-ePL-KRLS and ESM-ePL-KRLS (Alves et al. 2020).

Our major conclusions are as follows:

– The proposed model obtained the lowest errors (with the absence of significant outliers) in all simulations, suggesting that the forecasting model can predict complex behaviors with high accuracy.
– The proposed model obtained a low percentage of system-wide CPU utilization in all simulations, suggesting that it is appropriate to deal with Big Data.
– The forecast of fuel prices has applicability in the logistical context, for example. It can be used to optimize the delivery of a given cargo, defining the best days and places to fill up trucks in order to decrease the costs of delivery and therefore maximize profits.

This paper is organized as follows. Section 2 analyzes quantitatively and qualitatively historical S500 and S10 diesel oil prices and highlights their relevance in logistics. Section 3 introduces the variable step-size evolving Participatory Learning with Kernel Recursive Least Squares algorithm. Section 4 evaluates the proposed model. Section 5 concludes the research.

2 Problem formulation

Transport management is one of the main logistical processes and is also responsible for integrating the links in a supply chain, that is, the supply of inputs and raw materials between suppliers and production and the distribution of

final products between production and customers, respectively. Such management must be done efficiently and effectively to minimize costs and maximize the level of service.

According to Fleury et al. (2000), transportation costs correspond to approximately 60% of a company's logistical costs. Of these expenses, 40% represent fixed costs, such as depreciation and salaries for drivers, helpers, and mechanics. The remaining 60% represents variable costs, particularly maintenance, tolls, and fuel, in the case of a heavy truck used to transport cargo (Vilardaga 2007). Such research still estimates that the fuel is equivalent to about 58% of the variable costs and, consequently, 35% of the total expenses. In a country of continental dimensions such as Brazil, which primarily uses road networks to transport cargo and people, a computational tool that allows reliable forecasts on the prices for S500 and S10 diesel oil becomes a competitive differential for the logistics operator.

The fuel price dataset used to evaluate the model was obtained from the website of the National Agency of Petroleum, Natural Gas, and Biofuels (ANP) website (Agência Nacional do Petróleo 2020). Weekly resale and distribution prices at S500 and S10 diesel oil from December 31st, 2012 to May 30th, 2020 were considered. The main difference between diesel oils is related to the concentration of sulfur that each contains: S500 has 500 parts per million and S10 has 10 parts per million. This directly affects engine and environment, in which the S10 diesel oil has better ignition and combustion functioning, suffers less from corrosion of parts, and is less polluting.

The diesel oil market in Brazil is regulated by the National Petroleum Agency (ANP) and Federal Law 9.478/97 known as Petroleum Law. The price of diesel oil is composed of: 40% pure diesel, 11% biodiesel, 11% state taxes, 16% national taxes and 20% profit from distribution and resale (the final product must contain a mixture of 88% pure diesel and 12% biodiesel) according to Petrobras studies between May 24th, 2020 and May 30th, 2020 (Petrobras 2020).

A preliminary statistical analysis is presented in Table 1 and the behavior of historical prices can be observed in Fig. 1, requiring some additional information for a better
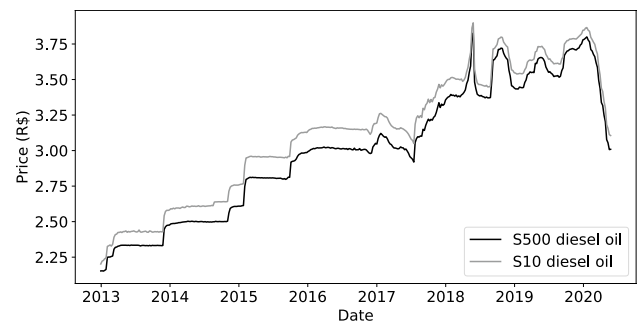
Table 1 Statistical analysis

	S500 diesel oil	S10 diesel oil
Data	386	386
Mean	3.008	3.124
Standard deviation	0.454	0.447
Minimum	2.152	2.202
First quartile	2.518	2.660
Median	3.014	3.155
Third quartile	3.391	3.504
Maximum	3.828	3.899

Fig. 1 Historical diesel oil prices

understanding. The first factor is the fuel price policy in Brazil. Until 2016, the price was controlled by Petrobras, which delayed the transfer of its international variations as a tool to control inflation. After several financial losses, the state-owned company changed this policy and the price started to be controlled by the barrel of oil in the international market. As their prices are due to several political and economic factors on a global scale, which has dynamic characteristics, the oscillation is more noticeable since 2017. The second factor is the successive increases in fuel prices, especially S500 diesel oil and S10 diesel oil which increased 8.383% and 7.755% in the first three weeks of May 2018, respectively. This resulted in the truckers strike (also called the diesel crisis) which lasted from May 21st, 2018 to May 30th, 2018 throughout the Brazilian territory which had several claims, including a reduction in the price of diesel. The third factor is COVID-19 pandemic. The international price of a barrel of oil plummeted, coming to cost -US$ 37.63 on April 20th, 2020, an event unprecedented in history. Its dynamic characteristic justifies the use of evolving fuzzy models.

3 Proposed model

The learning structure of variable step-size evolving Participatory Learning with Kernel Recursive Least Squares is shown in Figure 2.

It is an evolving fuzzy model based on Takagi–Sugeno–Kang rules (Takagi and Sugeno 1985; Sugeno and Kang 1988) . The rules configuration is "IF<antecedent>, THEN<consequent>" shown on Eq. 1. The antecedent is expressed by a linguistic variable and the consequent is expressed by a mathematical function. The main advantage of using fuzzy systems is its universal approximator propriety (Afravi and Kreinovich 2020). It is capable of approximating, with pre-specified precision, a general nonlinear continuous function on a compact set through the combination of local functions.
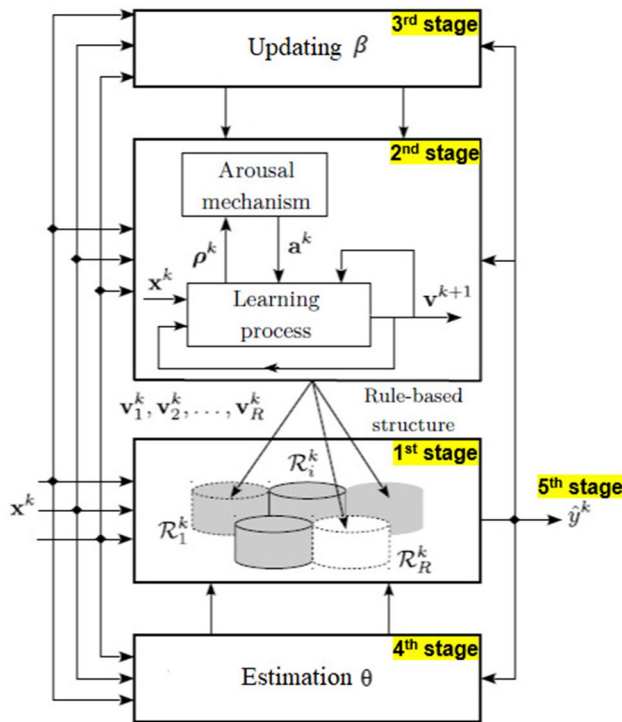
Fig. 2 VS-ePL-KRLS model

$$\mathcal{R}_i : \text{IF } x \text{ is } \mathcal{A}_i \text{ THEN } \hat{y}_i = f_i(x, \theta_i) \tag{1}$$

where $\mathcal{R}_i$ is the ith rule, $\mathcal{A}_i$ is the antecedent fuzzy set of the i-th rule, $x = [x_1, \ldots, x_m]^T \in \mathbb{R}^m$ is the input vector, $\theta_i = [\theta_{i1}, \ldots, \theta_{in_i}]^T \in \mathbb{R}^{n_i}$ is the consequent parameters vector of the ith rule, and $\hat{y}_i : \mathbb{R}^m \times \mathbb{R}^{n_i} \to \mathbb{R}$ is the local output of the ith rule.

The first stage of the model is the unsupervised fuzzy clustering algorithm proposed by Silva et al. (2005). It proved to be as efficient as Gustafson–Kessel (GK) Gustafson and Kessel (1979) and Modified Fuzzy K-Means (MFKM) (Gath et al. 1997), two major fuzzy clustering algorithms. The objective of this part is to adjust the number of fuzzy rules and their respective centers. The algorithm is initialized with a single rule ($R = 1$) whose center is the first input ($v_1^1 = x^1$). During the computational procedure, new rules can be added and existing rules can be updated or merged.

The second stage of the model is the evolving participatory learning proposed by Lima et al. (2010). It has a convergent conception with human learning (Yager 2004). In humans, the greater the prior knowledge in a given subject, the more amplified the learning will be. The computer learns in the same way in which each new information will be related to the existing ones and the greater the similarity, the greater the learning. This stage is divided into two steps.

In the first step is calculated the compatibility index (ρ_i^k) and the arousal index (a_i^k).

The compatibility index measures how well data and rules are compatible and it is represented by Eq. 2:

$$\rho_i^k = 1 - \frac{\left\| x^k - v_i^k \right\|}{m}. \tag{2}$$

where $\rho_i^k \in [0, 1]$ is the compatibility index of the ith rule at kth step, $x^k \in \mathbb{R}^m$ is the m-dimensional input vector at kth step, and $v_i^k \in [0, 1]^m$ is the center of the ith rule at kth step.

The arousal index evaluates whether a new rule must be added or an existing rule must be updated and it is represented by Eq. 3:

$$a_i^k = a_i^{k-1} + \beta^{k-1}\left(1 - \rho_i^k - a_i^{k-1}\right) \tag{3}$$

where $a_i^k \in [0, 1]$ is the arousal index of the ith rule at kth step and $\beta^k \in [0, 1]$ is parameter that controls the rate of change of arousal. The initialization values are $a_1^0 = 0$ and $\beta^0 = 0.18$.

The arousal index is compared with a parameter $\tau^k \in [0, 1]$. If $\left| a_i^k \right| > \tau^k$, a new rule is added. Otherwise, an existing rule with more compatibility is updated. At each step, the parameter is updated as $\tau^k = \beta^{k-1}$.

The center of the rule added $\left(v_R^{k+1}\right)$ is represented by Eq. 4:

$$v_R^{k+1} = x^k \tag{4}$$

The center of the rule updated $\left(v_i^{k+1}\right)$ is represented by Eq. 5:

$$v_i^{k+1} = v_i^k + \alpha\left(v_i^k\right)^{1-a_i^{k+1}}\left(x^k - v_i^k\right) \tag{5}$$

where $\alpha \in [0, 1]$ is the learning rate parameter.

In the second step is calculated the compatibility index between rules $\left(\rho_{ij}^k\right)$. It measures how redundant the rules are and evaluates whether an existing rule must be removed. It is represented by 6:

$$\rho_{ij}^k = 1 - \frac{1}{m}\sum_{l=1}^m \left| v_{il}^k - v_{jl}^k \right| \tag{6}$$

where $\rho_{ij}^k \in [0, 1]$ is the compatibility index between ith and jth rules at kth step for $i \neq j$.

The compatibility index between rules is compared with a parameter $\gamma^k \in [0, 1]$. If $\rho_{ij}^k > \gamma^k$, the least compatibility rule is removed. At each step, the parameter is updated as $\gamma^k = 1 - \beta^{k-1}$. The center of the rule removed $\left(v_j^k\right)$ is merged with the most compatibility rule center $\left(v_i^k\right)$ and it is represented by Eq. 7:

$$v_i^k = \frac{v_i^k + v_j^k}{2} \tag{7}$$

The third stage of the model presents the main innovation in relation to classic approaches. The β parameter, which controls the rate of change of arousal, is updated with the concept of Variable Step proposed by Harris et al. (1986). The normalized error is compared with a parameter $\bar{\gamma} \in [0, 1]$. If $|\tilde{e}^k| > \bar{\gamma}$, β increases and arousal index increases. Otherwise, β decreases and arousal index decreases. Equation 8 represents this procedure and the values α_{VS1}, α_{VS2} and $\bar{\gamma}$ are determined by heuristic.

$$\beta^{k+1} = \begin{cases} \frac{\beta^k}{\alpha_{VS1}}, & if \ |\tilde{e}^k| > \bar{\gamma} \\ \beta^k \, \alpha_{VS2}, & otherwise \end{cases} \tag{8}$$

where $\alpha_{VS1} \in \,]0, 1[$ and $\alpha_{VS2} \in \,]0, 1[$ are the factors that increases or decreases parameter β, respectively.

The fourth stage of the model is the estimation of the θ parameter proposed by Vieira et al. (2018). As shown on Eq. 1, each rule results in a local output $(\hat{y}_i)$ and it is represented by Eq. 9:

$$\hat{y}_i = \sum_{j=1}^{n_i} \theta_{ij}^{k-1} \, \kappa\left(d_{ij}^k, x^k\right)$$

$$\hat{y}_i = \sum_{j=1}^{n_i} \theta_{ij}^{k-1} \exp\left(-\frac{\left\|d_{ij}^k - x^k\right\|}{\sqrt{2}v_{ij}^k}\right)^2 \tag{9}$$

where θ_{ij}^{k-1} are the consequent parameters of the ith and jth rules at kth step, $\kappa(\cdot)$ is the Gaussian kernel function, d_{ij}^k is the element of dictionary, and v_{ij}^k is the kernel size.

The consequent parameters are estimated using an adaptive method known as Kernel Recursive Least Squares (KRLS). It is an extension of Recursive Least Squares (RLS), that increases the dimension of the input data space using a nonlinear function, generating simpler solutions. These values are calculated by determining the weight vector (ω) with the minimization cost function ($L(\omega)$). The optimization is represented by Eq. 10:

$$\min_{\omega} L(\omega) = \sum_{j=1}^{k} \left|y^j - \omega^T \phi^j\right|^2 + \lambda ||\omega||^2 \tag{10}$$

where $\lambda \in \left[10^{-5}, 10^{-2}\right]$ is the regularization parameter to avoid numerical instabilities at matrix (Haykin et al. 2009), ϕ^j is the high dimensional space at kth step, and y^j is the output at kth step.

The solution to the optimization of Eq. 10 is represented by Eq. 11:

$$\omega^k = \left[\lambda \mathbf{I} + \Phi^k \left(\Phi^k\right)^T\right]^{-1} \Phi^k Y^k$$

$$\omega^k = \Phi^k \left[\lambda \mathbf{I} + \left(\Phi^k\right)^T \Phi^k\right]^{-1} Y^k$$

$$\omega^k = \Phi^k \left[\lambda \mathbf{I} + \mathbf{K}^k\right]^{-1} Y^k \tag{11}$$

$$\omega^k = \Phi^k Q^k Y^k$$

$$\omega^k = \Phi^k \theta^k$$

where $\mathbf{I}$ is the identity matrix, $\Phi = \left[\phi^1, \ldots, \phi^k\right]$ is the high dimensional space vector, and $Y = \left[y^1, \ldots, y^k\right]^T$ is the output vector.

The kernel function can be used for generalizations of linear methods written in terms of internal products, represented by Eq. 12, in a Reproducing Kernel Hilbert Space (RKHS) (Scholkopf and Smola 2001), reducing its dimension and making it possible to solve Eq. 11.

$$\mathbf{K}^k = \left(\Phi^k\right)^T \Phi^k$$

$$\mathbf{K}^k = \begin{bmatrix} \kappa\left(x^1, x^1\right) & \ldots & \kappa\left(x^1, x^k\right) \\ \vdots & \ddots & \vdots \\ \kappa\left(x^k, x^1\right) & \ldots & \kappa\left(x^k, x^k\right) \end{bmatrix} \tag{12}$$

$$\mathbf{K}^k = \begin{bmatrix} \mathbf{K}^{k-1} & g^k \\ \left(g^k\right)^T & 1 \end{bmatrix}$$

where $\mathbf{K}^k$ is the kernel matrix and $g^k = \left[\Phi^{k-1}\right]^T \phi^k = \left[\kappa\left(x^1, x^k\right), \ldots, \kappa\left(x^{k-1}, x^k\right)\right]^T$.

The calculation of matrix $\left[\lambda \mathbf{I} + \mathbf{K}^k\right]$ is computationally expensive. Thus, Q^k variable is defined to recursively calculate it and expressed by Eq. 13:

$$Q^k = \left[\lambda \mathbf{I} + \mathbf{K}^k\right]^{-1}$$

$$Q^k = \left(r^k\right)^{-1} \begin{bmatrix} Q^{k-1} r^k + z^k \left(z^k\right)^T & -z^k \\ -\left(z^k\right)^T & 1 \end{bmatrix} \tag{13}$$

where $z^k = Q^{k-1} g^k$, $r^k = \lambda + \left[\phi^k\right]^T \phi^k - \left[z^k\right]^T g^k$, and $\tilde{e}^k$ is the error, that is, the difference between real value (y^k) and predicted value ($\hat{y}^k$).

The updated solution of consequent parameters is expressed by Eq. 14:

$$\theta^k = Q^k Y^k$$

$$\theta^k = \begin{bmatrix} \theta^{k-1} - z^k [r^k]^{-1} \tilde{e}^k \\ [r^k]^{-1} \tilde{e}^k \end{bmatrix} \tag{14}$$

where θ^k is the consequent parameters at k-th step.

It is important to note that the size of the kernel matrix increases quadratically in relation to the amount of data and, consequently, Q^k increases in the same way. This implies more and more expensive operations over time. To get around this situation was used a sparcification procedure as in Vieira et al. (2018) because it significantly decreases the required simulation time and memory, and also increases

the generalization ability Scholkopf and Smola (2001). A local dictionary $\left(\mathcal{D}^k\right)$ is defined for each rule and the dictionary belonging to the rule with the highest compatibility is altered which means that only the subset of the most relevant samples are considered for the update kernel matrix and parameters vector.

In this model, the novelty criteria Richard et al. (2008) were used because it compactly represents knowledge in each local dictionary and presents computationally low cost at the same time. The technique consists of calculating the minimum distance (ψ) between input and each element at the dictionary with Eq. 15 and two error scenarios: one with the new sample at local dictionary (ε) and other without the new sample at the local dictionary (ϵ). If $\psi < \delta$, the sample and all elements of the local dictionary are coherent and are not added to the local dictionary. Otherwise, the sample and all elements of the local dictionary are not coherent and are added to the local dictionary if $\varepsilon < \epsilon$. The value of $\delta = \frac{v_{ij}^k}{10}$ was calculated by heuristic method Richard et al. (2008), where v_{ij}^k is the kernel size for element d_{ij}^k nearer x^k.

$$\psi = \min_{(\forall d_{ij} \in \mathcal{D}_i^k)} \left\| x^k - d_{ij}^k \right\| \tag{15}$$

where ψ is the minimum distance between input and each element in the dictionary.

The process for choosing an appropriate value of kernel size is complex because a value too large can result in all similar data and a value too small can result in all distinct data. In this model, the recursive Levenberg–Marquardt algorithm Ngia et al. (1998) is used. It is a non-linear optimization method that is an intermediate version between gradient and Newton and it has the better convergence properties among the three algorithms. Each element v_{ij}^k in the vector of kernel parameters $\left(v_i^k = \left[v_{i1}^k, ..., v_{in_i}^k \right] \right)$ is associated with an element d_{ij}^k in the vector of local dictionary $\mathcal{D}_i^k = \left[d_{i1}^k, ..., d_{in_i}^k \right]$ with the final objective minimizing the local error function. This value is initialized with 0.5 for a new rule and Eq. 16 for a existing rule.

$$v_i^k = v_i^{k-1} + P_i^k \nabla_i^k \tilde{e}_i^k \tag{16}$$

where $P_i^k = \left[P_i^{k-1} - \frac{P_i^{k-1} \nabla_i^k [\nabla_i^k]^T P_i^{k-1}}{1 + [\nabla_i^k]^T P_i^{k-1} \nabla_i^k} \right]$,

$$\nabla_i^k = \Lambda_i^k \begin{bmatrix} \theta_{i1}^{k-1} - \frac{\left\| x^k - d_{i1}^k \right\|^2}{(v_{i1}^{k-1})^3} k(x^k, d_{i1}^k) \\ \vdots \\ \theta_{in_1}^{k-1} - \frac{\left\| x^k - d_{in_1}^k \right\|^2}{(v_{in_1}^{k-1})^3} k(x^k, d_{in_1}^k) \end{bmatrix} \text{and}$$

$\Lambda_i^k \in [0, 1]$ is the normalized activation degree of the ith fuzzy rule.

Algorithm 1: VS-ePL-KRLS model

Input: $x^k, k = 1, \ldots, n$
Output: $\hat{y}^k, k = 1, \ldots, n$
1 **Initialization:** $\alpha = 0.01, \beta^0 = 0.18, \gamma^0 = 0.18,$
   $\tau^0 = 0.82, \sigma = 0.05, \lambda = 10^{-4}, v_0 = 0.5$
2 **for** $k = 2, 3, \ldots, n$ **do**
3   **for** $i = 1, 2, \ldots, R$ **do**
4   *Calculate compatibility:*
5       $\rho_i^k = 1 - \frac{\left\| x^k - v_i^k \right\|}{m}$
6   *Calculate arousal:*
7       $a_i^k = a_i^{k-1} + \beta^{k-1} \left(1 - \rho_i^k - a_i^{k-1} \right)$
8   *Calculate local output:*
9       $\hat{y}_i^k = \sum_{j=1}^{|\mathcal{D}_i|} \theta_{ij}^{k-1} \kappa\left(d_{ij}^k, x^k\right)$
10   *Calculate parameter controls rate of change of arousal:*
11   **if** $|\tilde{e}^k| > \bar{\gamma}$ **then**
12       $\beta^k = \frac{\beta^{k-1}}{\alpha_{VS1}}$
13   **else**
14       $\beta^k = \beta^{k-1} \alpha_{VS2}$
15   **if** $\beta^k < IL$ **then**
16       $\beta^k = IL$
17   **if** $\beta^k > SL$ **then**
18       $\beta^k = SL$
19    $\tau^k = \beta^{k-1}$
20    $\gamma^k = 1 - \beta^{k-1}$
21   **if** $\min |a^k| > \tau$ **then**
22   *Calculate added center:*
23       $v_R^{k+1} = x^k$
24   *Initialize local dictionary with sample value:*
25       $\mathcal{D}_i^k = x^k$
26   *Calculate kernel size:*
27       $\nu_{R1}^k = 0.5$
28   *Calculate parameter consequent:*
29       $\theta_i^k = \left[\lambda + \kappa(x^k, x^k) \right]^{-1} y^{k-1}$
30   **else**
31   *Calculate updated center:*
32       $v_i^{k+1} = v_i^k + \alpha \left(v_i^k\right)^{1-a_i^{k+1}} \left(x^k - v_i^k\right)$
33   *Calculate minimum distance:*
34       $\psi = \min_{(\forall d_{ij} \in \mathcal{D}_i^k)} \left\| x^k - d_{ij}^k \right\|$
35   *Calculate errors:*
36       $\tilde{e}_i^k = \left| y_i^k - \hat{y}_i^k \right|$
37   **if** $\psi \geq \delta$ & $\varepsilon < \epsilon$ **then**
38   *Add sample at local dictionary:*
39          $\mathcal{D}_i^k = \mathcal{D}_i^k \cup x^k$
40   *Calculate kernel size:*
41       $\nu_{R1}^k = 0.5$
42   *Calculate parameter consequent:*
43       $\theta_i^k = \begin{bmatrix} \theta_i^{k-1} - z_i^k [r_i^k]^{-1} \tilde{e}_i^k \\ [r_i^k]^{-1} \tilde{e}_i^k \end{bmatrix}$
44   **for** $i, j = 1, 2, \ldots, c, i \neq j$ **do**
45   *Calculate centers compatibility:*
        $\rho_{ij}^k = 1 - \frac{1}{m} \sum_{l=1}^m \left| v_{il}^k - v_{jl}^k \right|$
46   **if** $\rho_{ij}^k > \gamma$ **then**
47   *Calculated merged center:*
48          $v_i^k = \frac{v_i^k + v_j^k}{2}$
49   *Calculate global output:*
50    $\underline{\hat{y}}^k = \sum_{i=1}^R \hat{y}_i \Lambda_i$

The fifth stage of the model is the global output, that is, the forecast value is the weighted average of local outputs considering individual rules contributions, resulting in the non-linear nature of the model. This variable is represented by Eq. 17:

$$\hat{y} = \sum_{i=1}^{R} \hat{y}_i \Lambda_i \tag{17}$$

4 Experimental results

To evaluate the effectiveness of the method proposed in this work, the models presented in previous sections are applied in the estimation of the S500 and S10 diesel oil, at the Brazilian level, for biweekly and monthly horizons. The training data (from January 2013 to December 2018) and test data (January 2019 to May 2020) were divided into 80% and 20%, respectively.

The parameters of ePL-KRLS, SM-ePL-KRLS, ESM-ePL-KRLS and VS-ePL-KRLS are defined as in Vieira et al. (2018) Alves et al. (2020): $\alpha = 0.01$, $\beta^0 = 0.18$, $\gamma^0 = 0.18$, $\tau^0 = 0.82$, $\sigma = 0.05$, $\lambda = 10^{-4}$ and $v_0 = 0.5$. The results are showed at Table 5.

The forecasting evaluation was measured with three error metrics: Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Non-Dimensional Index Error (NDEI). The biggest difference between them is that RMSE penalizes large errors and benefits small errors (between 0 and 1). These metrics are expressed by Eqs. 18, 19 and 20, respectively.

$$RMSE = \sqrt{\frac{1}{n} \sum_{k=1}^{n} \left(y^k - \hat{y}^k\right)^2} \tag{18}$$

where y^k is the real value and $\hat{y}^k$ is the predicted value

$$MAE = \frac{1}{n} \sum_{k=1}^{n} \left|y^k - \hat{y}^k\right| \tag{19}$$

where y^k is the real value and $\hat{y}^k$ is the predicted value

$$NDEI = \frac{1}{n} \sum_{k=1}^{n} \left|\frac{y^k - \hat{y}^k}{y^k}\right| \tag{20}$$

where y^k is the real value and $\hat{y}^k$ is the predicted value.

The computational performance evaluation was measured with a float that represents the current system-wide CPU utilization as a percentage. The test was realized in a computer with Intel(R) Core(TM) Processor i3-7020U CPU @ 2.30GHz 2.30 GHz, 4.00 GB installed memory (RAM) (with 3.87 GB usable) and 64-bit Operating System.

Table 2 Comparison between VS-ePL-KRLS and ePL-KRLS

Fuel	Horizon	MGN	p value	Observation
S500	Biweekly	-3.225	1.89×10^{-3}	H_0 **rejected**
	Monthly	-5.894	1.08×10^{-7}	H_0 **rejected**
S10	Biweekly	-3.595	5.85×10^{-4}	H_0 **rejected**
	Monthly	-7.094	6.97×10^{-10}	H_0 **rejected**

This approach was statistically validated with Morgan Granger Newbold (MGN) Test Granger and Newbold (2014) to evaluate if the accuracy between models are equivalent or not. MGN test is represented by Equation 21 and follows a student's t distribution with $n-1$ degrees of freedom.

$$MGN = \frac{\hat{\varrho}}{\sqrt{\frac{1-\hat{\varrho}}{n-1}}} \tag{21}$$

where $\hat{\varrho}$ is the correlation coefficient (Pearson coefficient) between sum and difference of errors for a time series with a size equal to an n. The null hypothesis is that two forecast error variances are equivalent.

As can be seen on Table 5, the proposed model showed the best prediction accuracy among evolving models for all error metrics. This result was statistically validated on Tables 2, 3 and 4 through the rejection of the null hypothesis for all comparisons between VS-ePL-KRLS and other models, indicating that two forecast error variances are not equivalent to a significance level of 5%. The computational performance, calculated through medium among 50 measurements of the percentage of current system-wide CPU utilization, has better results for VS-ePL-KRLS and ePL-KRLS models. The number of final rules of the proposed model

Table 3 Comparison between VS-ePL-KRLS and SM-ePL-KRLS

Fuel	Horizon	MGN	p value	Observation
S500	Biweekly	-2.819	6.20×10^{-3}	H_0 **rejected**
	Monthly	-3.876	2.30×10^{-4}	H_0 **rejected**
S10	Biweekly	-3.430	9.96×10^{-4}	H_0 **rejected**
	Monthly	-5.484	5.69×10^{-7}	H_0 **rejected**

Table 4 Comparison between VS-ePL-KRLS and ESM-ePL-KRLS

Fuel	Horizon	MGN	p value	Observation
S500	Biweekly	-3.068	3.02×10^{-3}	H_0 **rejected**
	Monthly	-5.901	1.05×10^{-7}	H_0 **rejected**
S10	Biweekly	-3.624	5.33×10^{-4}	H_0 **rejected**
	Monthly	-6.063	5.39×10^{-8}	H_0 **rejected**

Fig. 3 S500 diesel oil forecast for the biweekly horizon

Fig. 4 S500 diesel oil forecast for the monthly horizon

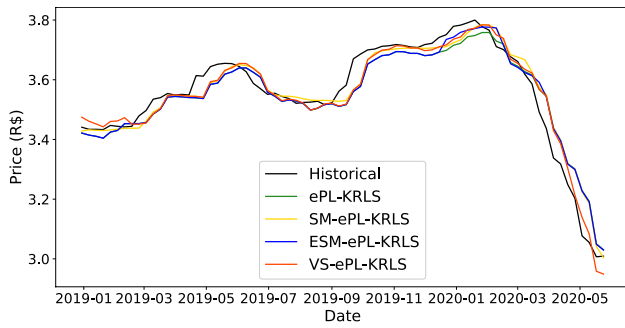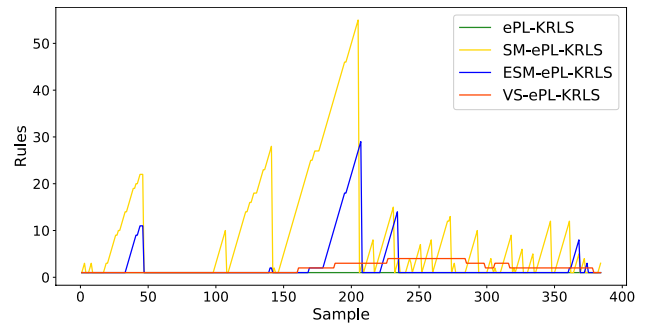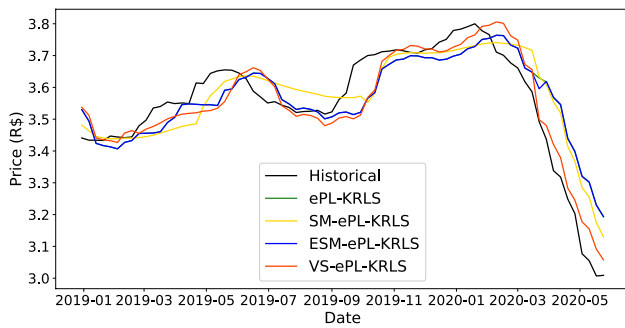**Fig. 5** S10 diesel oil forecast for the biweekly horizon

Fig. 6 S10 diesel oil forecast for the monthly horizon

Fig. 7 S500 diesel oil rules evolution for the biweekly horizon

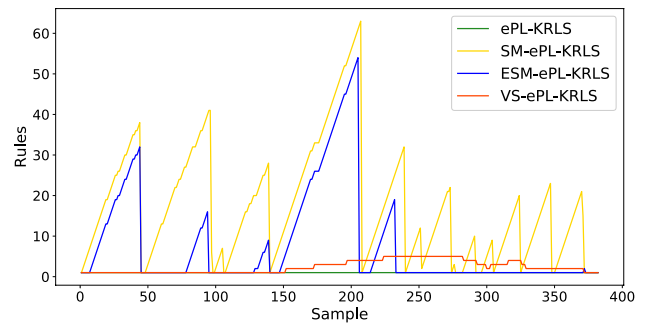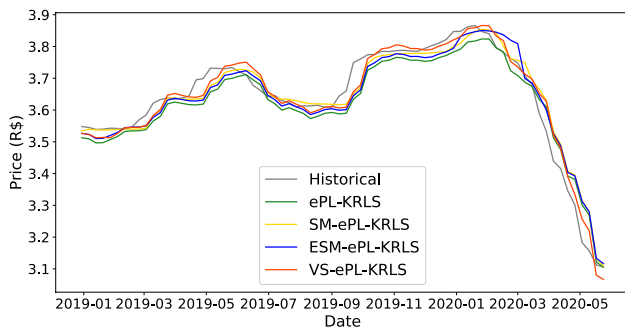**Fig. 8** S500 diesel oil rules evolution for the monthly horizon

was satisfactory. Figs. 3, 4, 5 and 6 show the predictions and Figs. 7, 8, 9 and 10 show the rules evolution.

5 Conclusions

In this paper, a new forecasting model is suggested to deal with the problem of fuel price forecasting: the Variable Step-Size evolving Participatory Learning with Kernel Recursive Least Squares, VS-ePL-KRLS. This model was tested for prediction of S500 and S10 diesel oil weekly prices in the context of Brazil for biweekly and monthly horizons and it appears as a better choice to integrate a decision support tool to assist the operational, tactical, and strategic logistic planning.

The evaluation of this model was measured in terms of error, the average number of rules, and computational complexity. The model proposed has better accuracy (supported by MGN test) and lower computational cost when compared to benchmarks evolving models suggested in the literature. Another benefit of the introduced models is that their structure makes the knowledge process continuous and more adaptable as the data changes than its evolving counterparts.
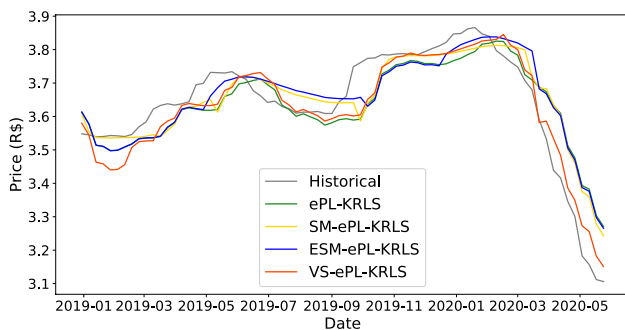
Table 5 Results

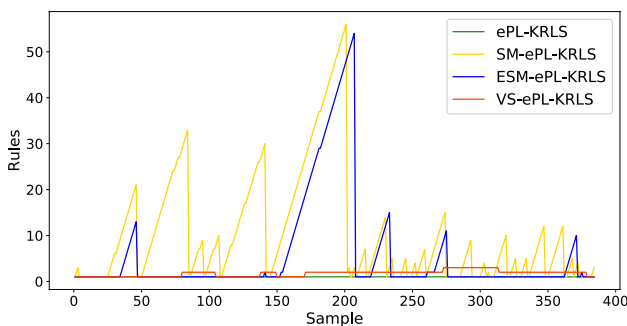| Fuel	Horizon	Algorithm	$\bar{\gamma}$	α_{VS1}	α_{VS2}	Average Number of Rules	RMSE	MAE	NDEI	%CPU
*S500	biweekly	ePL-KRLS	–	–	–	**1**	0.05061	0.03870	0.11239	13.82 ± 4.74
		SM-ePL-KRLS	0.010	–	–	9	0.04753	0.03368	0.10556	16.07 ± 5.57
		ESM-ePL-KRLS	0.014	–	–	3	0.04959	0.03717	0.11013	**13.14 ± 3.03**
		VS-ePL-KRLS	0.003	0.89	0.79	2	**0.04024**	**0.03066**	**0.08936**	13.70 ± 4.32
	monthly	ePL-KRLS	–	–	–	**1**	0.09526	0.07033	0.21304	14.35 ± 5.46
		SM-ePL-KRLS	0.028	–	–	17	0.08720	0.06772	0.19502	18.05 ± 4.59
		ESM-ePL-KRLS	0.019	–	–	8	0.09509	0.07008	0.21266	15.44 ± 5.09
		VS-ePL-KRLS	0.007	0.85	0.80	3	**0.06461**	**0.05110**	**0.14449**	**13.64 ± 4.61**
S10	biweekly	ePL-KRLS	–	–	–	**1**	0.04781	0.03932	0.10802	13.28 ± 3.32
		SM-ePL-KRLS	0.008	–	–	11	0.04238	0.02920	0.09575	17.07 ± 7.39
		ESM-ePL-KRLS	0.015	–	–	6	0.04411	0.03321	0.09966	15.77 ± 5.28
		VS-ePL-KRLS	0.001	0.97	0.84	2	**0.03486**	**0.02638**	**0.07877**	**13.19 ± 3.03**
	monthly	ePL-KRLS	–	–	–	**1**	0.08728	0.06722	0.19885	**13.44 ± 3.25**
		SM-ePL-KRLS	0.019	–	–	15	0.08056	0.05994	0.18355	18.29 ± 6.94
		ESM-ePL-KRLS	0.026	–	–	5	0.08339	0.06620	0.19000	14.22 ± 2.62
		VS-ePL-KRLS	0.003	0.74	0.45	4	**0.05972**	**0.04823**	**0.13606**	14.84 ± 5.55

Fig. 9 S10 diesel oil rules evolution for the biweekly horizon

Fig. 10 S10 diesel oil forecast for the monthly horizon

References

Afravi M and Kreinovich V (2020) Fuzzy systems are universal approximators for random dependencies: a simplified proof. In: Decision making under constraints, pp. 1–5, Springer

Agência Nacional do Petróleo GNeB (2020) Série histórica do levantamento de preços e de margens de comercialização de combustíveis

Alves KSTR, Hell M, Cyrino Oliveira FL, de Aguiar EP (2020) An enhanced set-membership evolving participatory learning with kernel recursive least squares applied to thermal modeling of power transformers. Electric Power Syst Res 184:106334

Angelov PP, Filev DP (2004) An approach to online identification of takagi-sugeno fuzzy models. IEEE Trans Syst Man Cybern Part B (Cybern) 34(1):484–498

Angelov P, Filev DP, Kasabov N (2010) Evolving intelligent systems: methodology and applications, vol. 12. Wiley

Chiu SL (1994) Fuzzy model identification based on cluster estimation. J Intell Fuzzy Syst 2(3):267–278

de Aguiar EP, Fernando MDA, Vellasco MM, Ribeiro MV (2017) Set-membership type-1 fuzzy logic system applied to fault classification in a switch machine. IEEE Trans Intell Transp Syst 18(10):2703–2712

Engel Y, Mannor S, Meir R (2004) The kernel recursive least-squares algorithm. IEEE Trans Signal Process 52(8):2275–2285

Evans JB, Xue P, Liu B (1993) Analysis and implementation of variable step size adaptive algorithms. IEEE Trans Signal Process 41(8):2517–2535

Fleury PF, Wanke P, Figueiredo KF (2000) Logística empresarial: a perspectiva brasileira. Editora Atlas SA

Gath I, Iskoz AS, Van Cutsem B (1997) Data induced metric and fuzzy clustering of non-convex patterns of arbitrary shape. Pattern Recogn Lett 18(6):541–553

Granger CWJ, Newbold P (2014) Forecasting economic time series. Academic Press

Gustafson DE, Kessel WC (1979) Fuzzy clustering with a fuzzy covariance matrix. In: 1978 IEEE conference on decision and control including the 17th symposium on adaptive processes, pp. 761–766, IEEE

Harris R, Chabries D, Bishop F (1986) A variable step (vs) adaptive filter algorithm. IEEE Trans Acous Speech Signal Process 34(2):309–316

Haykin SS et al. (2009) Neural networks and learning machines/simon haykin

Hyndman RJ, Athanasopoulos G (2018) Forecasting: principles and practice. OTexts

Kasabov N, Filev D (2006) Evolving intelligent systems: methods, learning, and applications. In: 2006 International symposium on evolving fuzzy systems, pp. 8–18, IEEE

Lima E, Hell M, Ballini R, Gomide F (2010) Evolving fuzzy modeling using participatory learning. Evolv Intell Syst Methodol Appl: 67–86

Lughofer E (2016) Evolving fuzzy systems–fundamentals, reliability, interpretability, useability, applications. In: Handbook on computational intelligence: Volume 1: fuzzy logic, systems, artificial neural networks, and learning systems, pp. 67–135, World Scientific

Maciel L, Vieira R, Porto A, Gomide F, Ballini R (2017) Evolving participatory learning fuzzy modeling for financial interval time series forecasting. In: 2017 Evolving and Adaptive Intelligent Systems (EAIS), pp. 1–8, IEEE

Ngia LS, Sjoberg J, Viberg M (1998) Adaptive neural nets filter using a recursive levenberg-marquardt search direction. In: Conference Record of Thirty-Second Asilomar Conference on Signals, Systems and Computers (Cat. No. 98CH36284), vol. 1, pp. 697–701, IEEE

Petrobras (2020) Diesel—composição de preços ao consumidor

Richard C, Bermudez JCM, Honeine P (2008) Online prediction of time series data with kernels. IEEE Trans Signal Process 57(3):1058–1067

Scholkopf B, Smola AJ (2001) Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press

Shafieezadeh-Abadeh S, Kalhor A (2016) Evolving Takagi–Sugeno model based on online Gustafson–Kessel algorithm and kernel recursive least square method. Evolv Syst 7(1):1–14

Silva L, Gomide F, Yager R (2005) Participatory learning in fuzzy clustering. In: The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ'05, pp. 857–861, IEEE

Škrjanc I, Iglesias JA, Sanchis A, Leite D, Lughofer E, Gomide F (2019) Evolving fuzzy and neuro-fuzzy approaches in clustering, regression, identification, and classification: a survey. Inform Sci 490:344–368

Sugeno M, Kang G (1988) Structure identification of fuzzy model. Fuzzy Sets Systems 28(1):15–33

Takagi T, Sugeno M (1985) Fuzzy identification of systems and its applications to modeling and control. IEEE Trans Syst Man Cybern 1:116–132

Vieira R, Gomide F, Ballini R (2018) Kernel evolving participatory fuzzy modeling for time series forecasting. In: 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), pp. 1–9, IEEE

Vilardaga V (2007) Choque de gestão no transporte rodoviário. Anuário de gestão de frotas, treinamento e pós-vendas 4:6–10

Yager RR (2004) Participatory learning: A paradigm for more human like learning. In: 2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No. 04CH37542), vol. 1, pp. 79–84, IEEE

Young PC (2012) Recursive estimation and time-series analysis: an introduction. Springer