A trading strategy based on BitCoin high and low prices: the role of an evolving fuzzy model for interval-valued time series forecasting

Leandro Maciel
Department of Business Administration
Faculty of Economics, Business, Accounting and Actuary, University of São Paulo, São Paulo, Brazil
leandromaciel@usp.br
ORCID: https://orcid.org/0000-0002-1900-7179

2023 IEEE International Conference on Fuzzy Systems (FUZZ) | 979-8-3503-3228-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/FUZZ52849.2023.10309799

***Abstract*—This paper evaluates the predictability of high and low BitCoin prices, and the profitability of a trading strategy based on these forecasts. As high and low prices can be seen as interval-value time series, an evolving fuzzy system to model and forecast interval data is suggested to capture time varying, nonlinear and uncertain dynamics of the cryptocurrency market. The model is composed of fuzzy functional rules in which its structure and functionality are updated as data are input. It is also designed to process interval-valued data, where high and low prices are used to represent the corresponding interval bounds. Antecedents of the rules are updated using participatory learning to cluster interval-valued time series, and consequents are computed using weighted recursive least squares based on intervals center and range. In addition to the evaluation of predictability through accuracy metrics, a simple trading strategy is constructed based on high and low prices forecasts to determine entry and exit signals, composing an economic criteria. Empirical results indicated that the fuzzy model is able to produce accurate forecasts of high and low prices of BitCoin, and a higher level of return adjusted by risk is achieved when these forecasts are used to perform a trading strategy in comparison with the competitive approaches.**

***Index Terms*—Interval time series, evolving fuzzy systems, BitCoin, forecasting, technical analysis.**

I. Introduction

Random walk and martingale efficient market theories state that the implementation of trading approaches based on past security prices are not able to outperform a simple buy-and-hold strategy [1]. On the other hand, technical analysts, or chartists, defend that trading rules using past information, called as technical analysis (TA), do yield higher returns than those that passively track the market. Indeed, works of [2]–[4] are examples that provide empirical evidence for TA[1]. [5], for instance, suggested an uncertainty reduction approach based on fuzzy logic that addresses two problems related to the uncertainty embedded in technical trading strategies: market timing and order size. Based on high-frequency exercises, authors showed that 'fuzzy technical indicators' dominate standard moving average technical indicators and filter rules for exchange rates, especially on high-volatility days.

[1]Technical analysis may be contrasted with fundamental analysis, which focuses on a company's financials rather than historical price patterns or stock trends.

Concerning high and low prices of equity shares, i.e., the maximum and minimum prices of an asset during a period of time, respectively, [6] contributed to TA literature with empirical evidence that these prices are predictable. Moreover, the research indicated that accurate forecasts of high and low prices can improve trading performance, supporting TA, as many trading rules based on this investment approach typically rely on resistance and support levels, which are related to high and low prices.

The modeling of high and low asset prices are particularly interesting not only for TA trading rules implementation. Traditional econometric time series models use single-valued time series data. In finance and economics, this approach is generally translated in the use of close/open asset prices to produce forecasts of future bahavior. However, it neglects the intraday price variability and relevant information is missed, compromising accuracy performance of the methods. To overcome this problem, the use of low and high prices is considered to construct financial interval-valued time series (ITS).

Several methods have been proposed in the literature to model and predict interval-valued time series, particularly linear models [7], [8] and threshold approaches [9]. Concerning machine learning methods, [10] proposed the interval Multi-Layer Perceptron (iMLP),and [11] presented a two stage forecasting process for interval-valued time series by predicting the intervals center and the radii in the realm of fuzzy systems.

This paper aims to contribute to the literature on TA by focusing on the predictability of high and low prices using an evolving fuzzy model designed to process interval-valued time series. In addition, empirical application concerns the digital currency market using actual data from BitCoin, the major traded cryptocurrency. The objective of this work is to address two main questions. A nonlinear, evolving fuzzy interval-valued technique is suitable to model these prices? And, do forecasts of high and low prices of BitCoin provide useful information to improve TA trading rules? To answer the first question, we compared the suggested method against different benchmarks for BitCoin high and low prices one-step-ahead forecasting. The approach suggested is an evolving interval fuzzy model that processes interval-valued input data as a stream, updating its structure and functionality as new data arrives. Its structure accounts for data time-varying, nonlinear and uncertain dynamic, typical in financial time series. It is composed by fuzzy functional rules where antecedents (cluster structure) are identified by participatory learning clustering, and the consequents updated using weighted recursive least squares, both learning methods translated to model ITS. Finally, to address the second question, i.e. do forecasts of high and low prices of BitCoin provide useful information for TA trading rules, a simple TA trading strategy is considered using high and low forecasts to determine entry and exit signals. One must notice that it is also a contribution to the empirical analysis of a relevant and highly volatile digital coin currency, the BitCoin, which demands accurate forecasters and more precise risk measures.

After this brief introduction, this work is organized as follows. The interval evolving fuzzy model proposed is described in Section II. Section III presents the empirical analysis for BitCoin high and low prices modeling and forecasting, in addition to the analysis of a trading rule using these price predictions. Finally, Section IV concludes the work and suggests topics for future research.

II. Evolving fuzzy modeling approach for interval-valued time series

A. Interval-valued time series

An interval can be represented by $[x] = [x^L, x^U] \in \mathcal{K}_c(\mathbb{R})$, where $\mathcal{K}_c(\mathbb{R}) = \{[x^L, x^U] : x^L, x^U \in \mathbb{R}, x^L \leq x^U\}$ is the set of closed intervals of the real line $\mathbb{R}$, and x^L and x^U are the lower and upper bounds of the interval $[x]$. The interval $[x]$ can also be denoted by a two-dimensional vector $[x] = [x^L, x^U]^T$.

When dealing with interval data, operations must be extended from traditional arithmetic to interval arithmetic. The arithmetic operations proposed by [12] are considered in this work. They are defined by:

$$\begin{aligned}
[x] + [y] &= [x^L + y^L, x^U + y^U] \\
[x] - [y] &= [x^L - y^U, x^U - y^L] \\
[x] \cdot [y] &= [\min\{x^L y^L, x^L y^U, x^U y^L, x^U y^U\}, \\
&\quad \max\{x^L y^L, x^L y^U, x^U y^L, x^U y^U\}] \\
[x] / [y] &= [x] \cdot (1/[y]), \; 1/[y] = [1/y^U, 1/y^L] \quad (1)
\end{aligned}$$

The center or mid-point of an interval-valued $[x]$, denoted as x^c is calculated as $x^c = \dfrac{x^L + x^U}{2}$, and the range (or radii) of $[x]$, x^r is $x^r = \dfrac{x^U - x^L}{2}$. Based on the center and the range of an interval-valued datum $[x]$, its lower and upper bounds can be recovered as $x^L = x^c - x^r$ and $x^U = x^c + x^r$, respectively.

The definition of the distance between two intervals is required in this paper, as it is a measure of their (dis)similarity, an important aspect when clustering the interval-valued data using participatory learning. This work considers the Hausdorff-Pompeiu distance for intervals [13], [14]. Denoting $dH([x],[y])$ as the distance between $[x]$ and $[y]$, it is computed as:

$$dH([x],[y]) = \max\left\{|x^L - y^L|, |x^U - y^U|\right\}. \quad (2)$$

Finally, a sequence of interval-valued variables in consecutive time steps t, $t = 1, 2, \ldots, N$, denoted by $\{[x]^t\} = \{[x]^1, [x]^2, \ldots, [x]^N\}$ is defined as an interval-valued time series (ITS), where N is the sample size or the number of intervals in the time series, and $[x]^t = [x^{L,t}, x^{U,t}]$. Evolving fuzzy system for interval-valued data (eFS$_I$) processes ITS as streams using the interval modeling framework described in this subsection. In the following, eFS$_I$ constructs and identification are discussed.

B. eFS$_I$ structure

eFS$_I$ for interval-valued data modeling is composed by a fuzzy rule base of interval fuzzy rules. Each fuzzy rule has an antecedent part identifying the state of the interval-valued input variable, and a consequent part specifying the corresponding interval-valued output variable. The model is defined by functional interval fuzzy rules, as proposed by [15] but extended for interval data. The interval fuzzy rule base in composed by c fuzzy rules of the form:

$$\mathcal{R}_i: \text{ IF } [\mathbf{x}] \text{ is } \mathcal{M}_i \text{ THEN } [y_i] = [\beta_{i,0}] + [\beta_{i,1}][x_1] + \ldots + [\beta_{i,n}][x_n], \quad (3)$$

where $\mathcal{R}_i$ is the i-th fuzzy rule, $i = 1, 2, \ldots, c$, c is the number of fuzzy rules in the rule base, $[\mathbf{x}] = ([x_1], [x_2], \ldots, [x_n])^T$ the vector of inputs, $[x_j] = [x_j^L, x_j^U] \in \mathcal{K}_c(\mathbb{R})$, $j = 1, \ldots, n$, and $[\beta_{i,l}] = [\beta_{i,l}^L, \beta_{i,l}^U] \in \mathcal{K}_c(\mathbb{R})$ are interval-valued parameters of the rule consequent, $l = 0, 1, \ldots, n$. $\mathcal{M}_i$ is the fuzzy set of the antecedent whose membership function is $\mu_i([\mathbf{x}]) : \mathcal{K}_c(\mathbb{R}^p) \to [0, 1]$, and $[y_i] = [y_i^L, y_i^U] \in \mathcal{K}_c(\mathbb{R})$ is the output of the i-th rule.

The output of the model, $[y]$, is computed as the weighted average of the interval rules in (3) as the fuzzy inference mechanism:

$$[y] = \sum_{i=1}^{c} \left(\frac{\mu_i([\mathbf{x}])[y_i]}{\sum_{j=1}^{c} \mu_j([\mathbf{x}])} \right) = \sum_{i=1}^{c} \lambda_i [y_i], \quad (4)$$

where $\lambda_i = \dfrac{\mu_i([\mathbf{x}])}{\sum_{j=1}^{c} \mu_j([\mathbf{x}])}$ is the normalized degree of activation of the i-th rule. Notice that the membership degree $\mu_i([\mathbf{x}])$ of datum $[\mathbf{x}]$ is given by:

$$\mu_i([\mathbf{x}]) = \left[\sum_{h=1}^{c} \left(\frac{\sum_{j=1}^{n} \left(\max\left\{|x_j^L - v_{i,j}^L|, |x_j^U - v_{i,j}^U|\right\}\right)}{\sum_{j=1}^{n} \left(\max\left\{|x_j^L - v_{h,j}^L|, |x_j^U - v_{h,j}^U|\right\}\right)} \right)^{\frac{2}{(m-1)}} \right]^{-1}, \quad (5)$$

where m is a fuzzification parameter (in this paper $m = 2$), and $[v_{i,j}] = [v_{i,j}^L, v_{i,j}^U] \in \mathcal{K}_c(\mathbb{R})$ is the cluster center of the i-th cluster/rule, $j = 1, \ldots, n$ and $i = 1, \ldots, c$.

eFS$_I$ learning is based on two steps: antecedents and consequents learning. Learning antecedents is related to clustering the interval-valued data space, and estimation of local models parameters to consequents identification.

C. eFS$_I$ antecedents learning

eFS$_I$ antecedent rules identification is performed using the participatory learning fuzzy clustering algorithm extended to handle interval-valued data, as proposed by [16]. The interval data set $[\mathbf{X}] = \{[\mathbf{x}_1], \ldots, [\mathbf{x}_N]\}$ is partitioned in c fuzzy subsets, $2 \leq c \leq N$, where N is the number of samples. Intervals $[\mathbf{x}_j]$ bounds are assumed to be normalized using min-max operator:

$$x^B_{norm} = \frac{x^B - \min\{x^L, x^U\}}{\max\{x^L, x^U\} - \min\{x^L, x^U\}}, \tag{6}$$

where B denotes either the lower bound L, or the upper bound U of the interval.

When using participatory learning for clustering the interval-valued data, it is assumed that the current cluster structure plays a key role in the learning process when new data is input. The compatibility of the data with the current structure directly influences cluster structure self-organization, which suggests the need for model revision based on the compatibility of the new data with the current cluster structure, defined by the cluster centers.

Cluster structure is defined by cluster centers, denoted by $[\mathbf{V}] = [[\mathbf{v}_1], \ldots, [\mathbf{v}_c]]$, $[\mathbf{v}_i] = ([v_{i,1}], \ldots, [v_{i,n}])^T$, $[v_{i,j}] = [v^L_{i,j}, v^U_{i,j}] \in \mathcal{K}_c([0,1])$, $j = 1, \ldots, n$ with $i = 1, \ldots, c$. The aim of participatory learning is the construction of model structure, i.e. defining $[\mathbf{V}]$ from inputs $[\mathbf{x}]^t \in [0,1]^n$, $t = 1, \ldots$. Notice that $[\mathbf{x}]^t$ is used as a vehicle to learn about $[\mathbf{V}]$.

The contribution of $[\mathbf{x}]^t$ to the learning process is evaluated based on its compatibility to the current cluster structure $[\mathbf{V}]^{t-1}$, i.e. the learning process is participatory. The compatibility $\rho^t_i \in [0,1]$ of input $[\mathbf{x}]^t$ with the cluster center $[\mathbf{v}_i]^{t-1}$ of $[\mathbf{V}]^{t-1}$, $i = 1, \ldots, c$ is obtained as:

$$\rho^t_i = 1 - \tfrac{1}{n}\textstyle\sum_{j=1}^n \left(\max\left\{|x^{L,t}_j - v^{L,t-1}_{i,j}|, |x^{U,t}_j - v^{U,t-1}_{i,j}|\right\}\right). \tag{7}$$

If cluster center $[\mathbf{v}_i]^{t-1}$ is the most compatible with $[\mathbf{x}]^t$, that is, $i = \arg\max_{j=1,\ldots,c}\{\rho^t_j\}$, then it is updated as follows:

$$[\mathbf{v}_i]^t = [\mathbf{v}_i]^{t-1} + G^t_i([\mathbf{x}]^t - [\mathbf{v}_i]^{t-1}), G^t_i = \alpha\rho^t_i \tag{8}$$

where $\alpha \in [0,1]$ is the basic learning rate. Notice that (8) can be viewed as a form of exponential smoothing modulated by the compatibility of data with the model structure, which is the nature of participatory learning.

If a sequence of input data with low compatibility with the current cluster structure indicates that the current model should be revised in front of new information, participatory learning uses an arousal mechanism to monitor the new cluster structure. The arousal mechanism acts as a reminder of when current cluster structure should be revised. A high arousal value indicates less confidence in how the current model fits recent input data, a complement of confidence [16]. The arousal $a^t_i \in [0,1]$ at step t is calculated as:

$$a^t_i = a^{t-1}_i + \beta(1 - \rho^t_i - a^{t-1}_i), \tag{9}$$

where $\beta \in [0,1]$ controls the rate of change of arousal. The closer β is to one, the faster the learning process senses compatibility variations.

If $a^t_i \geq \tau \in [0,1]$ for $i = 1, \ldots, c$, where τ is a threshold defined by the user, then a new cluster should be created, assigning the current data as its cluster center, that is, $[\mathbf{v}_{c+1}]^t = [\mathbf{x}]^t$. Otherwise, the center with the highest compatibility is updated to accommodate input the data using (8). The arousal mechanism (9) becomes part of the learning process by converting G^t_i of (8) in an effective learning rate:

$$G^t_i = \alpha(\rho^t_i)^{1-a^t_i}. \tag{10}$$

The clustering process using participatory learning also verifies the creation of redundant clusters that can be formed by using (8). A cluster i is redundant if its similarity with any other cluster h, $\rho^t_{i,h}$, is greater than or equal to a threshold $\lambda \in [0,1]$. The similarity between cluster centers i and h is found based on the corresponding distance of their centers, $dH([\mathbf{v}_i]^t, [\mathbf{v}_h]^t)$:

$$\rho^t_{i,h} = 1 - \tfrac{1}{n}\textstyle\sum_{j=1}^n \left(\max\left\{|v^{L,t}_{i,j} - v^{L,t}_{h,j}|, |v^{U,t}_{i,j} - v^{U,t}_{h,j}|\right\}\right). \tag{11}$$

If clusters i and h are declared redundant, then they are replaced by a cluster whose center is the average of their centers.

D. eFS_I consequents learning

eFS_I consequents learning comprises the estimation of the interval-valued parameters $[\beta_{i,0}], [\beta_{i,1}], \ldots, [\beta_{i,n}]$. Here, the classic weighted recursive least squares algorithm (wRLS) [17], [18] is considered. However, differently from [19], that computed parameters based on lower and upper bounds, center and range based-estimation are considered, as the evidence of its higher accuracy when modeling interval-valued data [9], [20]. Hence, parameters center $\beta^c_{i,l}$ and range $\beta^r_{i,l}$, with $i = 1, \ldots, c$ and $l = 0, 1, \ldots, n$, are estimated separately. Recall that the interval-valued parameter $[\beta]$ can be calculated as $\beta^L = \beta^c - \beta^r$ and $\beta^U = \beta^c + \beta^r$.

The output of eFS_I, in Eq. (4), can be expressed in terms of the input interval data center and range individually, namely, $y^{\{c,r\}} = \Lambda^T\Theta$ where $\Lambda = \left[\lambda_1\mathbf{x}^T_e, \lambda_2\mathbf{x}^T_e, \ldots, \lambda_c\mathbf{x}^T_e\right]^T$ is the fuzzily weighted input data, $\mathbf{x}_e = \left[1, x^{\{c,r\}}_1, x^{\{c,r\}}_2, \ldots, x^{\{c,r\}}_n\right]^T$ is the extended input, $\Theta = \left[\boldsymbol{\beta}^T_1, \boldsymbol{\beta}^T_2, \ldots, \boldsymbol{\beta}^T_c\right]^T$, and $\boldsymbol{\beta}_i = [\beta^{\{c,r\}}_{i,0}, \beta^{\{c,r\}}_{i,1}, \ldots, \beta^{\{c,r\}}_{i,n}]^T$. Superscript $\{c,r\}$ denotes either the center c, or the range (radii) r of the interval. The locally optimal error criterion wRLS is considered:

$$\min E^t_i = \min \sum_{k=1}^t \lambda_i \left(y^{\{c,r\},k} - (\mathbf{x}^k_e)^T\boldsymbol{\beta}^k_i\right)^2, \tag{12}$$

whose solution can be expressed recursively by [17]:

$$\boldsymbol{\beta}^{t+1}_i = \boldsymbol{\beta}^t_i + \Sigma^t_i\mathbf{x}^t_e\lambda^t_i\left(y^{\{c,r\},t}(\mathbf{x}^t_e)^T\boldsymbol{\beta}^t_i\right),\ \boldsymbol{\beta}^0_i = 0, \tag{13}$$

$$\Sigma^{t+1}_i = \Sigma^t_i - \frac{\lambda^t_i\Sigma^t_i\mathbf{x}^t_e(\mathbf{x}^t_e)^T\Sigma^t_i}{1 + \lambda^t_i(\mathbf{x}^t_e)^T\Sigma^t_i\mathbf{x}^t_e},\ \Sigma^0_i = \Omega I, \tag{14}$$

where Ω is a large number (typically $\Omega = 1{,}000$), and Σ is the dispersion matrix.

Expressions (13)-(14) individually estimate the center $\beta^c_{i,l}$ and the range $\beta^r_{i,l}$ of the interval-valued parameters $[\beta_{i,l}] = [\beta^L_{i,l}, \beta^U_{i,l}]$, $l = 0, 1, \ldots, n$, with $\beta^L = \beta^c - \beta^r$ and $\beta^U =$

$\beta^c + \beta^r$. Finally, the outputs of the fuzzy rules (3) at $t+1$ are obtained according to:

$$[y_i]^{t+1} = [\beta_{i,0}]^t + [\beta_{i,1}]^t [x_1]^t + \ldots + [\beta_{i,n}]^t [x_n]^t. \qquad (15)$$

with $i = 0, 1, \ldots, c$.

The overall model output $[y]^{t+1}$ at $t+1$ is the weighted average of the outputs $[y_i]^{t+1}$ computed using (4).

E. eFS$_I$ algorithm

The evolving fuzzy modeling approach for interval-valued data, eFS$_I$, updates model structure and functionality recursively, being memory efficient when processing stream data. This is very appealing when real-time applications are taken into account, as in performing investment decisions in high volatile markets such as the BitCoin, considered in this work. The steps of eFS$_I$ construction are provided in the following.

eFS$_I$ algorithm

choose parameters α, β, τ, and λ.
set $a_1^1 = 0$, $\Sigma_i^0 = \Omega I$, $[\beta_0^1] = [0, 0]$.
start cluster structure: $c = 1$, $[\mathbf{v}_1]^1 = [\mathbf{x}]^1$.
for $t = 2, 3, \ldots$ **do**
  read $[\mathbf{x}]^t$
  for $i = 1, \ldots, c$
    compute $\rho_i^t = 1 - dH([x], [y])$
    update $a_i^t = a_i^{t-1} + \beta(1 - \rho_i^t - a_i^{t-1})$
  end for
  if $a_i^t \geq \tau$ for $i = 1, \ldots, c$
    create a new cluster: $c = c + 1$
    $[\mathbf{v}_c]^t = [\mathbf{x}]^t$
    reset $a_i^t = 0$
  else
    most compatible cluster s: $s = \max_{j=1,\ldots,c} \{\rho_j^t\}$
    update $[\mathbf{v}_s]^t = [\mathbf{v}_s]^{t-1} + G_s^t([\mathbf{x}]^t - [\mathbf{v}_s]^{t-1})$,
  end if
  for $i = 1, \ldots, c-1$ and $h = i+1, \ldots, c$
    compute compatibility $\rho_{i,h}^t = 1 - \frac{1}{n} dH([\mathbf{v}_i]^t, [\mathbf{v}_h]^t)$
    if $\rho_{i,h}^t \geq \lambda$
      redefine $[\mathbf{v}_i]^t$ using centers i and h, and delete $[\mathbf{v}_h]^t$
      $c = c - 1$
    end if
  end for
  compute rule consequent parameters using the wRLS
  compute model output $[y]^{t+1}$
end for

III. Computational Experiments

This work evaluates the predictability of high and low prices of BitCoin using an evolving fuzzy modeling framework designed for interval-valued stream data. Performance of eFS$_I$ is compared against Autoregressive Integrated Moving Average (ARIMA), Multilayer Perceptron Neural Networks (MLP) and random walk (RW) models for one-step-ahead predictions.

A. BitCoin data

Data is comprised by the daily high and low prices of BitCoin (BTC) as the corresponding interval-valued time series (ITS). The sample covers the period from January 1, 2017 to December 31, 2019, for a total of 1,095 observations[2]. The data was divided into in-sample and out-of-sample sets. The in-sample data set contains data from 2017-2018, and the out-of-sample data contains the 2019 data. All evaluations reported concerns the out-of-sample set, as the in-sample set was used for models learning and validation. This approach is not necessary for eFS$_I$, as it processes stream data differently for the other counterparts, ARIMA, MLP and RW. These former methods produce interval forecasts from the individual lowest and highest daily BTC data, computed separately. On the other hand, eFS$_I$ do processes ITS naturally, considering the interval structure of the data.

In addition, the trading strategy considered in this work, detailed in the next section, uses high-frequency prices of BTC at the one-minute frequency. The sample of high frequency prices covers the period from July 1 00:00:00, 2019 to September 30 23:59:00, 2019 for a total of 132,480 observations[3].

B. Forecasting evaluation and trading strategy

Models performance is computed according to the mean absolute percentage error (MAPE) and the root mean squared error (RMSE), respectively, $\mathrm{MAPE}^B = \frac{100}{N} \sum_{t=1}^{N} \frac{|y^{B,t} - \hat{y}^{B,t}|}{y^{B,t}}$, $\mathrm{RMSE}^B = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y^{B,t} - \hat{y}^{B,t})^2}$, where the superscript B denotes either the lower bound L, or the upper bound U of the interval BTC prices, $[y]^t = [y^{L,t}, y^{U,t}]$ is the actual price interval, $[\hat{y}]^t = [\hat{y}^{L,t}, \hat{y}^{U,t}]$ is the forecast price interval at t, and N is the size of the out-of-sample data set.

Additionally, in practice, the direction of price change is as important as, sometimes more important than, the magnitude of the forecasting error [21]. Hence, it is also considered a measure of direction accuracy (DA) [22] as an alternative to compute models forecasting performance. DA is calculated as:

$$\mathrm{DA}^B = \frac{1}{N} \sum_{t=1}^{N} Z^{B,t}, \qquad (16)$$

with $Z^{B,t} = 1$ if $\left(\hat{y}^{B,t+1} - y^{B,t}\right)\left(y^{B,t+1} - y^{B,t}\right) > 0$, and $Z^{B,t} = 0$ otherwise.

Finally, a trading strategy is implemented based on the predicted high and low prices as an economic application of eFS$_I$. Using one-step-ahead forecasts of the high and low prices for the out-of-sample set using intraday data, price bands are constructed. High and low prices are linked to the notions of support and resistance.

The trading strategy evaluated in this work is the BTC intradaily trading, where the high and low prices predictions are used to obtain the forecast of the range (radii) of price bands. Following [6], from the high-low bands, we define buy and sell signals by comparing the intradaily evolution of the BTC price and the predicted daily range-based bands. During a given trading day, if the price crosses the high band, we consider it to be a sell signal. Conversely, if the price crosses the low band, we perceive this as a buy signal.

The range-based strategy uses high-frequency prices of BTC at the one-minute frequency. The results of the contrarian

[2]The data are available at https://coinmarketcap.com/. Sample begins in 2017 due the higher liquidity of BTC, and ends in 2019 to avoid the impact of extreme events such as the COVID-19 pandemic in 2020.

[3]Data are available in https://www.kaggle.com/mczielinski/bitcoin-historical-data.

strategy based on the range bands, for the evaluated models, are also compared with a buy-and-hold strategy, $B\&H$, which simply involves buying stock at $t = 1$ and selling it at $t = N$ (end of the sample). Comparisons of the trading strategies are made in terms of cumulative returns (r_C), as a measure of profitability, computed by $r_C = \left[\prod_{t=1}^{N}(1 + r_t)\right] - 1$, where $r_t = ln(p_t) - ln(p_{t-1})$ is the log-return, and p_t the price at instant t. As a measure of risk, the volatility of the investment approaches, σ, is computed as the returns standard deviation $\sigma = \sqrt{\frac{1}{T}\sum_{t=1}^{N}(r_t - \bar{r})^2}$, with $\bar{r}$ as the mean return of the trading strategy.

Finally, as investors make their decisions combining the joint evaluation of return and risk, the Sharpe Ratio (SR) of the strategies is calculated, with $\text{SR} = \frac{r_C}{\sigma}$. SR measures the profitability of an investment strategy by unit of risk.

C. Empirical results

eFS$_I$ forecasting ability is compared against ARIMA, MLP and RW models for BitCoin high and low prices prediction. Notice that ARIMA, MLP and RW are univariate time series techniques, i.e. an individual model is estimated for each interval bound (lower and upper). On the other hand, eFS$_I$ is an interval-valued method and produces interval bands (low and high prices) forecasts. In this paper, eFS$_I$ uses as input lagged values of the interval time series $[y]^t = [y^{L,t}, y^{U,t}]$. Hence, one-step-ahead forecasts, $[y]^{t+1} = [y^{L,t+1}, y^{U,t+1}]$, are produced as $[\hat{y}]^{t+1} = f_{eFS_I}\left([y]^t, [y]^{t-1}, [y]^{t-2}, \ldots, [y]^{t-l}\right)$, where l is the number of lagged values used as input, and $f_{eFS_I}(\cdot)$ encodes the eFS$_I$ modeling framework.

Models setting and training were performed concerning the in-sample data, from January 2017 to December 2018. Structures of ARIMA and MLP are different for lower (L) and upper (U) interval bounds, low and high prices, respectively. For both BitCoin low and high prices, an ARIMA(2,1,2) model was selected according to the Bayesian information criteria (BIC), i.e. data were integrated with order equals to unit, and the model considers two autoregressive components and two autoregressive moving average terms. Regarding the neural network models, the MLPs selected present a single hidden layer with 6 and 12 neurons for low and high BTC prices, respectively. These structures are associated with the lower values of the accuracy metrics for the in-sample set. For both interval bounds, MLPs inputs comprised four lagged values of the corresponding time series. Finally, simulations where performed to select eFS$_I$ control parameters associated with the lowest values of MAPE and RMSE, which are: $l = 5$, $\beta = 0.12$, $\tau = 0.52$, $\alpha = 0.04$ and $\lambda = 0.82$. eFS$_I$ was implemented in MATLAB, and ARIMA, MLP and RW were developed using software R with the use of the *forecast* package.

Models accuracy performance for low and high prices of BitCoin forecasting in terms of RMSE, MAPE and DA is presented in Table I. Best performance is highlighted in bold. For RMSE and MAPE, the lower the value the better the prediction method. On the other hand, high values of DA are associated with the best forecaster of prices direction. Notice that DA values for the RW are not shown as the method uses the actual price to produce one-step-ahead forecast $\left([\hat{y}]^{t+1} = [y]^t\right)$, i.e. the direction of price change can not be computed.

TABLE I
RMSE, MAPE AND DA PERFORMANCE FOR ONE-STEP-AHEAD FORECASTING OF DAILY BITCOIN LOW AND HIGH PRICES.

Metric	ARIMA	MLP	RW	eFS$_I$
Panel A: BTC low price				
RMSE	264.20	267.95	**262.12**	263.76
MAPE	2.0471	2.0762	**1.9950**	2.0283
DA	57.85	59.72	-	**61.17**
Panel B: BTC high price				
RMSE	323.35	326.29	**291.12**	293.72
MAPE	2.2315	2.2466	**1.9903**	2.0421
DA	52.61	55.19	-	**59.87**

In terms of accuracy, the results from RMSE and MAPE, for both BitCoin low and high prices forecasting, indicated that the naïve RW model outperforms all the competitors, which is associated with the lowest values of the metrics (see Table I). Overall, the rankings from best to worst, in terms of RMSE and MAPE, for both BTC ITS bounds are: RW, eFS$_I$, ARIMA and MLP. eFS$_I$ achieved the second best performance, superior than MLP and ARIMA. The higher accuracy of RW is in line with the well-known "Meese-Rogoff" [23] puzzle, which states that prediction models are not able to outperform random walk in out-of-sample forecasting for exchange rates. The results from Table I do support this conclusion for the case of digital currencies like the BitCoin. However, literature showed that this is valid only if forecasting accuracy is measured in terms of criteria that depend on the magnitude of the forecasting error only [24], [21].

From the point of view of the direction accuracy (DA), Table I indicates that eFS$_I$ outperforms ARIMA and MLP for price direction forecasting of both BTC low and high prices. This better performance of eFS$_I$ in forecasting directions of price changes is due to its evolving, continuous adaptation ability to capture price changes more accurately in time-varying environments such as in the BitCoin market, in which the level of volatility is considerably high. In addition, eFS$_I$ has the ability of capturing the patterns hiding in ITS as it takes into account the possible mutual dependency between the daily highs and lows of BTC prices. A simple proportion z-test, with the null hypothesis for the equality of DA between two competitive models, has indicated that eFS$_I$ is statistically superior than ARIMA and MLP, with p-values of 0.0152 and 0.0401 (0.0286 and 0.0452) when BTC low (high) prices direction forecast is concerned, respectively, at a 5% significance level. Notice that MLP performed better than ARIMA in terms of DA (see Table I).

Models performance was also evaluated through the implementation of a contrarian trading strategy using intraday BTC low and high prices forecasts, as discussed in subsection III-B. Table II reports the results of the trading strategy based on the daily range bands. Strategies were carried out using eFS$_I$, ARIMA, MLP forecasts, which are also compared with a simple buy-and-hold strategy (B&H). Table II indicates that the strategy based on the range, obtained from eFS$_I$ interval forecasting method, showed the higher cumulative return and the lower level of volatility, measured by the returns standard deviation, in comparison with the competitors. The B&H strategy achieved a negative cumulative return as the BTC prices have decreased in the associated period. In terms of return adjusted by risk, the Sharpe Ratio (SR) values showed that ARIMA and MLP have a similar trading performance, but both provided worse results than the eFS$_I$.

TABLE II
STRATEGY PERFORMANCES BASED ON PREDICTED BTC PRICE RANGES (LOW AND HIGH PRICES) IN TERMS OF CUMULATIVE RETURNS (r_c), VOLATILITY (σ) AND SHARPE RATIO (SR). B&H IS THE STRATEGY BASED ON BUY AT THE INITIAL DATE AND SELL AT THE FINAL DATE.

Metric	**ARIMA**	**MLP**	**eFS$_I$**	**B&H**
r_c	26.12	25.76	28.43	-28.66
σ	43.12	44.65	40.76	45.76
SR	0.61	0.58	0.70	-0.63

IV. CONCLUSION

This paper suggested an evolving fuzzy inference method for interval-valued time series modeling and forecasting, named eFS$_I$. The approach is based on functional fuzzy rules that continuously update its structure and functionality using interval stream input data. Based on BitCoin high and low prices, one-step-ahead forecasts from eFS$_I$ were compared with ARIMA, MLP and random walk models. eFS$_I$ overperformed the competitors when performance is measured by the direction of price changes, and when forecasts are used to perform a trading strategy. Future works include: comparisons of eFS$_I$ against alternative interval forecasting techniques, evaluation of accuracy using prediction metrics for interval-valued data and statistical tests for competitive forecasters, and the use of predictions to compute the volatility range as a tool for risk management.

ACKNOWLEDGEMENTS

This work was supported by the Brazilian National Council for Scientific and Technological Development (CNPq) under grant number 304456/2020-9, and by the Ripple Impact Fund, a donor advised fund of Silicon Valley Community Foundation, grant number 2018-196450(5855), as part of the University Blockchain Research Initiative, UBRI.

REFERENCES

[1] A. Ang and G. Bekaert, "Stock return predictability: is it there?" *Review of Financial Studies*, vol. 20, no. 3, pp. 651–707, 2007.

[2] L. Blume, D. Easley, and M. O'Hara, "Market statistics and technical analysis: the role of volume," *Journal of Finance*, vol. 49, no. 1, pp. 153–181, 1994.

[3] W. Brock, J. Lakonishok, and B. LeBaron, "Simple technical trading rules and the stochastic properties of stock returns," *Journal of Finance*, vol. 47, no. 1731-1764, 1992.

[4] A. W. Lo, H. Mamaysky, and J. Wang, "Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation," *Journal of Finance*, vol. 55, no. 4, pp. 1705–1770, 2000.

[5] N. Gradojevic and R. Gençay, "Fuzzy logic, trading uncertainty and technical trading," *Journal of Banking & Finance*, vol. 37, no. 2, pp. 578–586, 2013.

[6] M. Caporin, A. Ranaldo, and P. S. Magistris, "On the predictability of stock prices: A case for high and low prices," *Journal of Banking & Finance*, vol. 37, pp. 5132–5146, 2013.

[7] L. Billard and E. Diday, "Regression analysis for interval-valued data," in *Data Analysis, Classification and Related Methods: Proceedings of the 7th Conference of the IFCS*, 2000, pp. 369–374.

[8] E. Lima Neto and F. de Carvalho, "Constrained linear regression models for symbolic interval-valued variables," *Computational Statistics & Data Analysis*, vol. 54, no. 2, pp. 333–347, 2010.

[9] P. Rodrigues and N. Salish, "Modeling and forecasting interval time series with threshold models," *Advances in Data Analysis and Classification*, vol. 9, no. 1, pp. 41–57, 2015.

[10] A. Roque, C. Maté, J. Arroyo, and A. Sarabia, "iMLP: Applying multi-layer perceptrons to interval-valued data," *Neural Processing Letters*, vol. 25, no. 2, pp. 157–169, 2007.

[11] W. Degang, S. Wenyan, and W. Pedrycz, "A two stage forecasting approach for interval-valued time series," *Journal of Intelligent & Fuzzy Systems*, vol. 35, no. 2, pp. 2501–2512, 2018.

[12] R. Moore, R. Kearfott, and M. Cloud, *Introduction to interval analysis*. Philadelphia: SIAM Press, 2009.

[13] D. Huttenlocher, G. Klanderman, and W. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 9, pp. 850–863, 1993.

[14] B. Li, Y. Shen, and B. Li, "A new algorithm for computing the minimum hausdorff distance between two point sets on a line under translation," *Information Processing Letters*, vol. 106, no. 2, pp. 52–58, 2008.

[15] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Transactions on Systems Man and Cybernetics*, vol. SMC-15, no. 1, pp. 116–132, 1985.

[16] L. Maciel, R. Ballini, F. Gomide, and R. R. Yager, "Participatory learning fuzzy clustering for interval-valued data," in *Proceedings of the 16th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2016)*, Eindhoven, The Netherlands, 2016, pp. 1–8.

[17] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[18] P. Angelov, "Evolving Takagi-Sugeno fuzzy systems from data streams (eTS+)," in *Evolving intelligent systems: Methodology and applications*, P. Angelov, D. Filev, and N. Kasabov, Eds. Hoboken, NJ, USA: Wiley & IEEE Press, 2010, pp. 21–50.

[19] L. Maciel, R. Ballini, and F. Gomide, "Adaptive fuzzy modeling of interval-valued stream data and application in cryptocurrencies prediction," *Neural Computing and Applications*, vol. https://doi.org/10.1007/s00521-021-06263-5, 2021.

[20] J. Arroyo, R. Espínola, and C. Maté, "Different approaches to forecast interval time series: A comparison in finance," *Computational Economics*, vol. 27, no. 2, pp. 169–191, 2011.

[21] K. Burns and I. Moosa, "Enhancing the forecasting power of exchange rate models by introducing nonlinearity: Does it work?" *Economic Modelling*, vol. 50, pp. 27–39, 2015.

[22] J. Hamilton, *Time Series Analysis*. Princeton, New Jersey: Princeton University Press, 1994.

[23] R. Meese and K. Rogoff, "Empirical exchange rate models of the seventies: Do they fit out of sample?" *Journal of International Economics*, vol. 14, no. 1–2, pp. 3–24, 1983.

[24] I. Moosa and K. Burns, "The unbeatable random walk in exchange rate forecasting: Reality or myth?" *Journal of Macroeconomics*, vol. 40, pp. 69–81, 2014.