DOI 10.33111/nfmte.2022.157

FUZZY TIME SERIES FORECASTING USING SEMANTIC ARTIFICIAL INTELLIGENCE TOOLS

Andrii Bielinskyi

State University of Economics and Technology
16 Medychna Str., Kryvyi Rih, 50005, Ukraine
ORCID: 0000-0002-2821-2895, E-mail: krivogame@gmail.com

Vladimir Soloviev

Kryvyi Rih State Pedagogical University
54 Gagarin Ave., Kryvyi Rih, 50086, Ukraine
ORCID: 0000-0002-4945-202X, E-mail: vnsoloviev2016@gmail.com

Victoria Solovieva

State University of Economics and Technology
16 Medychna Str., Kryvyi Rih, 50005, Ukraine
ORCID: 0000-0002-8090-9569, E-mail: vikasolovieva2027@gmail.com

Halyna Velykoivanenko

Kyiv National Economic University named after Vadym Hetman
54/1 Peremogy Ave., Kyiv, 03680, Ukraine
ORCID: 0000-0001-6326-3965, E-mail: ivanenko@kneu.edu.ua

This study investigates the application of Fuzzy Time Series (FTS) methods in forecasting the Bitcoin market. FTS methods have gained significant attention due to their simplicity, adaptability, forecasting precision, and computational efficiency. They generate interpretable representations of time series patterns, enabling knowledge transfer, auditability, reusability, and upgradability. The study specifically focuses on time-invariant rule-based FTS techniques, namely the conventional First-Order FTS (Song and Chen) and Weighted First-Order FTS (Yu) models. The research rigorously evaluates and compares the predictive performance of these methods across a range of accuracy metrics. Additionally, the article expands the understanding and application of FTS methods in cryptocurrency forecasting. Through comprehensive experimental evaluations and statistical analyses, it uncovers insights into the strengths, limitations, and potential areas for improvement of these FTS approaches. By highlighting their comparative accuracy and computational efficiency, the research contributes to the existing studies and provides practical recommendations for researchers and practitioners in the cryptocurrency domain.

Keywords: *fuzzy time series, fuzzy set, fuzzy logical relationships group, fuzzy time series forecasting, Bitcoin, cryptocurrency market*

JEL Classification: C53, C63, G10

Introduction

Accurately predicting future events or phenomena holds immense significance in numerous scientific, engineering, medical, economic, and other domains. The process of forecasting serves as a potent tool for drawing meaningful inferences about future conditions and values based on historical and present information. As a fundamental objective of time series analysis, forecasting encompasses a broad spectrum of algorithms, ranging from traditional statistical methods to more flexible and adaptive soft computing approaches.

Conventional methods like moving average, regression, or exponential moving average encounter several challenges. First of all, these methods are constrained by the assumption of linearity, which may not hold in complex real-world systems. Additionally, these methods lack the flexibility to incorporate linguistic variables or expert opinions, limiting their applicability in capturing nuanced aspects of the data [10, 39, 47, 59].

To overcome these limitations, alternative approaches have emerged, offering more robust and versatile forecasting methodologies. Soft computing techniques, for instance, leverage the power of fuzzy logic, neural networks, or evolutionary algorithms to handle nonlinear relationships, accommodate linguistic variables, and integrate expert knowledge into the forecasting process. By harnessing the strengths of these advanced methods, researchers and practitioners can enhance the accuracy and effectiveness of their forecasts, enabling more informed decision-making and planning in various fields.

Since its inception in 1965 by Zadeh [62], fuzzy set theory has yielded substantial advancements in both theoretical underpinnings and practical applications. The primary impetus behind the development of fuzzy set theory lies in its ability to offer a formal, robust, and quantitative framework for handling the inherent vagueness present in human knowledge, as conveyed through natural languages [22]. It is precisely due to this capability that fuzzy set theory has emerged as a potent and cutting-edge tool, distinct from conventional methodologies.

By embracing fuzzy set theory, researchers can leverage its unique features to tackle the complexities associated with uncertain and imprecise information. The underlying philosophy of fuzzy set theory is rooted in capturing and modeling the inherent fuzziness and ambiguity prevalent in human cognition and linguistic expressions. This enables the incorporation of subjective judgments, qualitative assessments, and linguistic terms into the forecasting process, which may be challenging or even unattainable with traditional quantitative methodologies alone. By embracing and harnessing the potential of fuzzy set theory, researchers can effectively bridge the gap between human perception and mathematical modeling, providing a more comprehensive and accurate representation of real-world phenomena.

With the increasing application of fuzzy set theory in diverse domains such as decision-making, planning, economics, logical reasoning, systems theory, artificial intelligence, control theory, and others, scholars have extensively explored and presented realistic problems alongside corresponding methodologies. Unlike conventional time series analysis, which operates on real numbers, fuzzy set-based time series $F(t)$ comprises a collection of time-indexed fuzzy linguistic terms $f_1(t), f_2(t), \dots, f_n(t)$, offering significant capabilities to handle the inherent uncertainty and ambiguity associated with data collection.

Recent years have witnessed a growing interest in fuzzy logic approaches to financial analysis and forecasting. Fuzzy Time Series (FTS) methods, as multiple studies have highlighted their favorable accuracy. FTS methods, belonging to the realm of soft computing, facilitate the generation of data-driven, non-parametric, easily interpretable models for analyzing and forecasting time series. Notably, these methods possess several advantages, including computational efficiency, simplicity, interpretability, and cost-effectiveness, without the need for extensive datasets as required by statistical and deep learning models.

Analysis of recent research and publications

For financial time series prediction, artificial intelligence tools are often used, which include as machine learning methods [16-18, 29, 34, 36, 44, 56, 57], the principle of operation of which is based on identifying patterns without the use of expert knowledge about the processes under study, as well as AI tools, which are based on priori established information about existing functional dependencies [33, 39, 40]. Noteworthy is the original approach to modeling financial and economic indicators [30, 32, 40, 41] using Kohonen self-organizing maps. Here, based on key features, vectors of input data describing the modelling objects or processes are initially segmented into a number of clusters, each of which combines observations according to their similarity. Next, for each cluster its own forecast model is built, or the cluster corresponds to a certain class of changes in the indicator under study (for example, decline, flat, growth, etc.). Ivanchenko and Vashchaiev [28] presented a study on the dynamic evolutionary processes of trophic relations among enterprise populations. Bifurcation analysis of mergers and acquisitions dynamic modes, utilizing Kohonen self-organizing map for pattern recognition, categorizes enterprise life cycle phase portraits into five classes.

Kmytiuk and Majore [31] used recurrent neural network models for forecasting, specifically Elman and Jordan architectures, in order to level out the negative impact of high levels of nonlinearity and nonstationarity, noise, the presence of irregular trends, jumps, and anomalous emissions inherent in time series. Olivares et al. [42] enhanced the Neural Basis Expansion Analysis (NBEATS) by incorporating exogenous factors, resulting in the NBEATSx method. The proposed deep learning neural network offers interpretability, allowing the structural decomposition of time series and visualization of the impact of trend, seasonal components, and interactions with exogenous factors. The work [45] introduces TimeGrad, an autoregressive denoising diffusion model for multivariate probabilistic time series forecasting on real-world data sets with thousands of correlated dimensions.

The paper [18] proposes a solution to the problem of the short-term forecasting of trend movements of stock indices using deep learning approaches (in particular, RNNs based on cells of LSTM, CNNs, stacking CNN-RNN, and MLP as benchmark). Here, CNN allows to automatically extracting features and hidden patterns from the data, which are pass to the input of the block LSTM for making prediction.

Another deep learning architecture was developed in the study [64] to address issues of long short-term forecasting by introducing Informer, an efficient transformer-based model with more efficient self-attention mechanism and generative-style decoder for faster inference of long time-series sequences.

As mentioned above, the second type of AI tools that are used to forecast financial indices, based on priori established information about existing functional dependencies. This empirical knowledge forms the basis of various kinds of expert systems, and primarily, as regards artificial intelligence, fuzzy inference systems. Since such AI systems are based on formalized representations of linguistically specified knowledge and data, they are called semantic.

An example of the implementation of this approach is presented in article [39], where the forecast for the development of stock market indicators was obtained using fuzzy logic tools, the rule base of which was formed on the basis of Elliott wave theory. In the paper [40] a fuzzy inference system was built for forecasting foreign direct investment, the optimization of which was carried out on the basis of a set of characteristic indicators of countries selected as a result of clustering using the Kohonen map into one homogeneous group.

An equally interesting concept is multivariate FTS, which allow us to capture multidimensional fuzzy relationships between each of the variables under consideration. Chen and Chang [12] proposed a new method for multivariate forecasting, employing fuzzy clustering and rule interpolation techniques. The method involves constructing training samples based on variation rates, utilizing fuzzy C-means clustering to form fuzzy rules, assigning weights to these rules based on input observations, and employing a multiple fuzzy rules interpolation scheme for predicting output. Bitencourt et al. [7] considered the difficulties of multivariate FTS when dealing with multivariable non-stationary datasets. They presented new approach, which considers projecting the original high-dimensional data into low-dimensional representation with subsequent FTS application.

The paper [23] introduced a hybrid approach for analyzing seasonal FTS using novel partial high-order bivariate fuzzy time series forecasting model. The study [24] presented a method for determining fuzzy relationships without relying on fuzzy logic relation tables. Instead, a feedforward artificial neural network is used for this purpose. Thesis [14] introduced scalable FTS methods for point, interval, and probabilistic forecasting of mono and multivariate time series, considering one to many steps ahead. The discussion covers parameters, hyper-parameters, and fine-tuning alternatives.

The work [21] proposed a methodology to address the issue of high-dimensional data by projecting it into a low-dimensional embedding space using Kohonen self-organizing maps. Subsequently, the weighted multivariate FTS method is employed for rule discovery and forecasting. Singh and Borah [48] introduced a hybrid model merging fuzzy time series theory with artificial neural networks. To enhance forecasting accuracy, a “Re-Partitioning Discretization” approach is employed to determine effective interval lengths in historical time series data. High-order fuzzy relationships are utilized to improve forecasting precision. The model avoids using current state’s fuzzified values for prediction, enhancing predictive skill. The proposed model incorporates an artificial neural network-based architecture to defuzzify time series values.

Paper [55] employed fuzzy sets and a multivariate FTS method for predicting future prices of the Chinese stock index. By integrating traditional FTS models with the rough set method, the data is discretized using the fuzzy C-mean algorithm. Rules are initially formed in mature modules of the rough set and then refined using data mining algorithms. The study [61] applied neural networks to enhance FTS forecasting, introducing bivariate models for improved accuracy.

Some of our previous studies considered approaches of complex systems science for the construction of indicators-precursors of crash phenomena [4]. Here [2, 5, 6, 49, 51], we have demonstrated the possibility of constructing indicators of crash events on the basis of the theories of complex networks, non-extensive statistics, and information entropy approaches in combination with fuzzy logic theory. We have shown that corresponding indicators behave as precursors of crisis events in cryptocurrency and stock markets. In one of our studies [50], we have applied chaos theory methods to the Dow Jones Industrial Average Index.

The focus of the study [3] was to measure the varying irreversibility of the stock market. Therefore, we have applied measures of irreversibility (asymmetry) such as Guzik index, Porta index, Costa index, indices that are based on complex networks measures and permutation patterns, and, moreover, we have included multiscale time irreversibility index to estimate the degree of irreversibility in financial market. We have shown that the corresponding measures can serve as indicators-precursors of financial crisis states. The disadvantage of our approach is that we can only use indicators to measure the complexity of the dynamics we had before or now. We can make inferences about what might happen in the future, but the methods mentioned above are not responsible for extrapolating the price of a cryptocurrency or a stock index. Therefore, in this paper we would like to start by analyzing old but time-tested approaches that may have synergy with our indicators in the long run.

In the subsequent sections of this article, we delve into the fundamental principles and methodologies of fuzzy set theory, elucidating its key concepts and applications in the context of forecasting. Through empirical investigations and comparative analyses of Bitcoin market forecasting, we demonstrate the efficacy and advantages of employing fuzzy set theory as a powerful and versatile tool for predictive modeling.

Before we begin to explain the key idea of FTS and predictive algorithms based on them, we will introduce and explain several key concepts that form the basis of FTS. More specifically, we will start with the definitions of the universe of discourse (UoD) and the membership function.

Universe of discourse

The universe of discourse encompasses the entire range of values observed in the data: $U = [\min(Y), \max(Y)]$.

Although this study focuses on time-invariant methods and stationary time series Y, it is crucial to acknowledge that the testing data U may exhibit slight variations compared to the training data. These variations can arise due to factors such as outliers, which cannot be entirely disregarded even in stationary processes. To accommodate these fluctuations and facilitate the fuzzification process in the forecasting procedure, it is common practice to include a confidence margin that exceeds the upper and lower bounds. In this study, the UoD is established as $U = [\min(Y) - 0.2 \cdot |\min(Y)|$, $\max(Y) + 0.2 \cdot |\max(Y)|]$, exceeding the boundaries by 20%.

By accounting for the fluctuations in the boundaries of the known data U, the forecasting model becomes more robust and adaptable, capable of handling unforeseen variations in the testing data. Consequently, the exceeded margins enhance the overall accuracy and resilience of the forecasting methodology.

Membership function

Once U has been defined, three hyperparameters will determine the creation of $\tilde{A}$: the number of partitions (fuzzy sets) k, the membership function μ, and the partitioning scheme. The membership function $\mu: U \rightarrow [0,1]$ defines how much a crisp value belongs to corresponding fuzzy set, in terms of the membership grade $[0,1]$. Some options of fuzzy membership functions are shown in Table 1.

Table 1

MOST COMMON FUZZY MEMBERSHIP FUNCTIONS

Type	Parameters	Definition
Triangular	a: lower bound, b: mid-point, c: upper bound	$\mu(x,a,b,c) = \begin{cases} 0 & \text{if} \quad x \leq a \\ \dfrac{x-a}{b-a} & \text{if} \quad a < x \leq b \\ \dfrac{c-x}{c-b} & \text{if} \quad b < x \leq c \\ 0 & \text{if} \quad c < x \end{cases}$
Trapezoidal	a: lower bound, b: top-left, c: top-right, d: upper bound	$\mu(x,a,b,c,d) = \begin{cases} 0 & \text{if} \quad x \leq a \\ \dfrac{x-a}{b-a} & \text{if} \quad a < x \leq b \\ 1 & \text{if} \quad b < x \leq c \\ \dfrac{d-x}{d-c} & \text{if} \quad c < x \leq d \\ 0 & \text{if} \quad d < x \end{cases}$
Gaussian	m: midpoint, σ: spread	$\mu(x,m,\sigma) = \exp\left(-\dfrac{(x-m)^2}{2\sigma^2}\right)$
Bell-shaped	m: midpoint, σ: spread	$\mu(x,m,\sigma) = \dfrac{1}{1+\left(\frac{x-m}{\sigma}\right)^2}$

A combination of trapezoidal and triangular functions has been also considered for modeling fuzzy time series [8, 39]. However, since the mentioned membership functions are composed of straight line segments, they are not smooth at the corner points specified by the parameters. Therefore, smooth and non-linear Gaussian and bell-shaped membership functions have been adopted as well, by considering midpoint and spread as their parameters.

It is worth notions that in comparison to the other factors, membership function has less effect on the forecasting accuracy, which will be demonstrated empirically later on this work, but the proper choice of μ can directly impact on the readability and explainability of the model. In addition, the parameters of the membership functions can change when tuning the model on the training set.

In addition to the mentioned membership functions, which are more common, some references developed different FTS methods applying other types of fuzzy set membership functions, including picture fuzzy sets and picture fuzzy clustering [25], type-2 fuzzy sets [1, 38, 43], hesitant differential fuzzy sets [20], non-stationary fuzzy sets [13, 20] and so on.

Fuzzy time series idea

This section provides a concise overview of fuzzy time series, drawing insights from [52, 53]. One fundamental distinction between fuzzy time series and conventional time series lies in the nature of their values. While conventional time series consist of real numbers, fuzzy time series incorporate fuzzy sets [62], which can be thought of as classes with flexible boundaries.

The definition of fuzzy time series can be formulated as follows. Consider a conventional time series $Y = \{y(t), t = 1,2,3, \dots\}$ defined on the universe of discourse U. Upon UoD we define k fuzzy sets f_j, $j = 1, \dots, k$, each of which is described by its own membership function μ_{T_j}. The group of fuzzy sets f_j constitutes the linguistic variable $\tilde{A}$ [63], which, by language elements, such as words and phrases, describes the base variable (in our case, “Bitcoin exchange rate”). So, each fuzzy set can be denoted by the linguistic term T_j, $j = 1, \dots, k$. Terms can represent, for example, “high”, “medium”, “low” Bitcoin exchange rate, or “rise”, “flat”, “fall” if the linguistic variable is “Change in Bitcoin exchange rate”.

Fuzzy time series $F(t)$ is the collection of fuzzified sets of Y: $F(t) = \{f_j(t), j = 1, \dots, k\}$, $t = 1,2,3, \dots$ According to [52], $F(t)$ can be understood as a linguistic variable with own fuzzy values $f_j(t)$, $j = 1, \dots, k$. Since across the time the values of $F(t)$ can be different, $F(t)$, as a set of fuzzy sets, is a function of time t. Moreover, the universes of discourse can be different at different times [52].

So, the distinguishing mark of fuzzy time series is that their values are fuzzy sets, while the values of ordinary time series are real numbers.

The linguistic variable $\tilde{A}$ and a knowledge representation of the time series dynamics compose the FTS model. The main components of the **training method** of fuzzy model are listed below:

1. *Partitioning*: The most important process of the training is executed, the partitioning. This process is responsible to split the universe of discourse U into k fuzzy sets f_j with correspondent membership functions μ_{T_j}. Each fuzzy set f_j characterized by own linguistic term T_j of the linguistic variable $\tilde{A}$ used to describe Y. There are many ways that the partitioning can be performed. In this study, we partition UoD into sets with an equal length.

2. *Fuzzification*: The original time series data Y is transformed using the chosen membership function (Gaussian) into a linguistic representation, the fuzzy time series $F(t)$, where each $y(t) \in Y$ is presented by a set of fuzzy sets $F(t) = \{f_j(t), j = 1, \dots, k\}$. When forming a fuzzy time series, to the value of the original time series $y(t)$ is assigned a special formed vector A_j of membership values at discrete points of the fuzzy set f_j with the largest value of the membership function $\mu_{T_j}(y(t))$ at time *t*:

$$y(t) \to A_j \mid \mu_{T_j}(y(t)) = \max\{\mu_{T_1}(y(t)), \dots, \mu_{T_k}(y(t))\}. \qquad (1)$$

3. *Knowledge Extraction and Representation*: The knowledge extraction phase involves the generation of the knowledge model, which represent temporal fuzzy transitions in the studied series.

After defining the linguistic variable $\tilde{A}$ and learning the FTS model, it becomes possible to generate a forecast $\hat{y}(t+1)$. The essential components of the **forecasting process** are outlined below:

1. *Fuzzification*: Following the training procedure, the process of fuzzification is applied to the initial data using the same framework. This step, detailed in the following sections, involves converting the data into fuzzy sets for subsequent determination of the most likely transitions at the next point in time to some level of the predicted indicator, described by the corresponding linguistic term.

2. *Rule Matching*: The aim of the rule matching procedure is to produce $F(t+1)$ – candidate fuzzy sets (and other additional information, as weights) to represent the future crisp value $y(t+1)$.

3. *Defuzzification*: Once $F(t+1)$ is generated, the defuzzification process is employed to transform the fuzzy representation into a precise numerical estimate, denoted as $\hat{y}(t+1)$, which corresponds to the approximate value of $y(t+1)$.

The main procedure of model training and forecasts generation is presented schematically in Fig. 1.

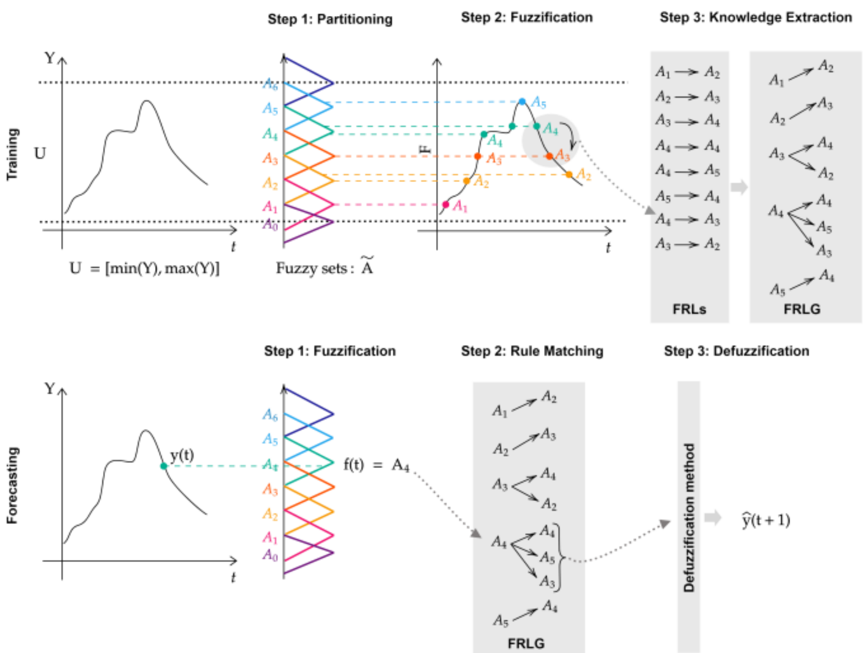

Fig. 1. Training and Forecasting Procedures for Time Invariant FTS [15]

Song and Chissom matrix-based method

In their seminal work, Song and Chissom [53] introduced the utilization of a Fuzzy Relationship Matrix to capture the temporal dynamics within the fuzzy time series. This approach involves grouping sequential pairs $F(t-1), F(t)$ into Fuzzy Logical Relationships (FLR). Each FLR represents a fuzzy rule that describes a specific temporal pattern, namely $f_i(t-1) \to f_j(t)$ or $A_i \to A_j$ (or, equivalently,

$F(t-1) \rightarrow F(t)$). Here, the Left Hand Side (LHS) of the rule, referred to as the precedent, denoted as A_i, represents the fuzzified historical value at time $t-1$. Conversely, the Right Hand Side (RHS) of the rule, known as the consequent, denoted as A_j, represents the fuzzified value of the predicted indicator at time t. Thus, the rule $A_i \rightarrow A_j$ can be interpreted as "if $y(t-1)$ is A_i, then $y(t)$ is A_j".

Suppose, that we can obtain fuzzy logical relationships: $A_1 \rightarrow A_1$, $A_1 \rightarrow A_2$, $A_2 \rightarrow A_3$, $A_3 \rightarrow A_4$, $A_4 \rightarrow A_6$, $A_6 \rightarrow A_6$, and $A_6 \rightarrow A_7$, where $A_1, A_2, \dots, A_7$ are the row vectors

$$A_i = \left\{\mu_{T_i}(y_1), \mu_{T_i}(y_2), \dots, \mu_{T_i}(y_7)\right\}, \qquad i = 1, \dots, 7, \tag{2}$$

which are the sets of membership functions of corresponding discrete fuzzy sets

$$f_i = \frac{\mu_{T_i}(y_1)}{y_1} + \frac{\mu_{T_i}(y_2)}{y_2} + \dots + \frac{\mu_{T_i}(y_7)}{y_7}, \qquad i = 1, \dots, 7, \tag{3}$$

where $\mu_{T_i}(y_j)$ is the membership function of set f_i, which indicates the degree of membership of y_j to f_i, denoted by the term T_i; y_j, $j = 1, \dots, 7$, are discrete points of the predicted variable $y(t)$ on the UoD [52].

Suppose A and B are row vectors of dimension k and $D = (d_{ij}) = A^T \times B$, where "×" is the Cartesian product of two fuzzy membership sets. The element d_{ij} of matrix D at row i and column j is defined as $d_{ij} = \min(A_i, B_j)$ for $i, j = 1, \dots, k$, where A_i and B_j are the i-th and j-th elements of A and B.

So, for proposed fuzzy logical relationships let $R_1 = A_1^T \times A_1$, $R_2 = A_1^T \times A_2$, $R_3 = A_2^T \times A_3$, $R_4 = A_3^T \times A_4$, $R_5 = A_4^T \times A_6$, $R_6 = A_6^T \times A_6$, $R_7 = A_6^T \times A_7$. Fuzzy relationship matrix can be defined in the following way:

$$R(t, t-1) = R = \bigcup_{i=1}^{7} R_i, \tag{4}$$

where R is a 7×7 matrix and $\cup$ is the union operator. Using R, the forecasting model can be defined as

$$A_j = A_i \circ R, \tag{5}$$

where A_i is, as an example, the Bitcoin closing price on day $t-1$, and A_j is the forecasted Bitcoin closing price on day t.

Chen rule-based method

Chen model is recognized as one of the conventional approaches known for its simplicity in calculations and improved forecasting accuracy [11]. To illustrate its usage, we present the following step-by-step procedure:

Step 1: Defining the UoD and intervals for observations.

Step 2: Defining fuzzy sets for observations.

Step 3: Fuzzifying observations.

In this step, each price value is mapped to a corresponding fuzzy set. Following the approach in [11, 54], a value $y(t)$ is fuzzified as A_j if the fuzzy set $f_j(t)$ exhibits the maximum degree of membership in the set of fuzzy sets $F(t)$ (1).

Step 4: Establishing FLRs.

To establish FLRs, we pick consecutive pairs of fuzzy sets: $(A_i \rightarrow A_j)$.

Step 5: Establishing fuzzy relationships.

FLRs sharing the same LHSs can be grouped into an ordered FLR group. This grouping is achieved by combining all the RHSs of the FLRs into the RHS of the FLR group [11]. For example, FLRs such as $A_i \rightarrow A_{j1}, A_i \rightarrow A_{j2}, \dots, A_i \rightarrow A_{jg}$ can be grouped into a fuzzy group $A_i \rightarrow A_{j1}, A_{j2}, \dots, A_{jg}$. As a result, repeated FLRs are counted only once [11, 52, 54].

Step 6: Forecasting.

Assuming $F(t-1) = A_i$, the forecasting procedure determines the value of $F(t)$ according to the following rules:

Rule 1: If A_i has no subsequent LHS, the forecasted value is equal to A_i.

Rule 2: If $A_i \rightarrow A_{j1}, A_{j2}, \dots, A_{jg}$, then the next value of $F(t)$ is equal to $A_{j1}, A_{j2}, \dots, A_{jg}$.

Step 7: Defuzzification.

Often, it is desirable to obtain numerical values instead of fuzzified values. If $F(t) = A_{j1}, A_{j2}, \dots, A_{jg}$, the defuzzified result is calculated as the arithmetic average of the midpoints of $A_{j1}, A_{j2}, \dots, A_{jg}$ [11]:

$$\hat{y}(t) = \frac{\sum_{p=1}^{g} m_{jp}}{g}, \tag{6}$$

where $\hat{y}(t)$ is the resulting forecast at time t, and m_{jp} is the midpoint of membership function μ_{jp} of a fuzzy set A_{jp}.

The application of Chen FLR group models provides a concise and easily interpretable depiction of the behavior exhibited by time series data through the use of fuzzy rules. This property holds the potential to benefit not only business experts but also researchers involved in knowledge extraction, as exemplified by Lee et al. [35]. However, beyond the enhanced interpretability, there exists another compelling reason to favor Chen model over the Song and Chissom approach: performance.

The size of the relation matrix in the Song and Chissom model increases proportionally to the number of partitions (obtained fuzzy sets) in the universe of discourse. This growth of the number of the fuzzy sets poses a significant increase in the dimensionality of the obtained FLR group, which can negatively impact the computational efficiency when dealing with large datasets in the forecasting process. In contrast, Chen FLR group models offer an advantage in this regard. By employing a compact representation and utilizing fuzzy rules, Chen models mitigate the curse of dimensionality, thereby alleviating the computational burden associated with forecasting tasks involving extensive datasets.

Thus, in addition to their human-readable representation, Chen FLR group models demonstrate superior performance in terms of computational efficiency, making them a favorable choice for forecasting applications, particularly when confronted with large datasets. By minimizing the impact of the curse of dimensionality, Chen models enable expedited and more practical forecasting, enhancing the overall effectiveness and scalability of the methodology.

Yu weighted-based rule method

The process of generating FLR groups from the fuzzified data within the FTS model presents two notable limitations: the loss of rule recurrence and the disregard for chronological order. Consequently, during the forecasting process, a highly recurrent pattern in the data holds the same significance as a unique occurrence pattern. Additionally, both newer and older patterns are assigned equal weight in the forecast.

Furthermore, the omission of repeated FLRs in the establishment of fuzzy relationships can have significant implications on the accuracy and reliability of the forecasting process. By neglecting the recurrence and chronological order of rules, valuable information regarding patterns and trends within the data may be overlooked, leading to suboptimal forecasting outcomes.

To address this limitation, it is crucial to develop approaches that preserve the recurrence and chronological order of rules within the FLR group generation process. By considering the repeated FLRs and their sequential arrangement, the forecasting model can capture and leverage the temporal dynamics and recurring patterns present in the data, resulting in improved forecasting performance.

In this section we describe old but efficient method [60] that takes into account the recurrence and chronological order of FLRs when constructing the FLR group. This approach aims to enhance the accuracy and effectiveness of the forecasting process by incorporating valuable information from both repeated and unique patterns, while appropriately considering the temporal aspects of the data. In previous studies [11, 14, 26, 27, 52, 54], the presence of repeated FLRs was simply overlooked when establishing fuzzy relationships. To elucidate this point, we refer to the FLR group [11] as an example:

$$\begin{array}{ll} (t=1) & A_1 \rightarrow A_2, \\ (t=2) & A_2 \rightarrow A_3 \\ (t=3) & A_3 \rightarrow A_3, \\ (t=4) & A_3 \rightarrow A_4, \\ (t=5) & A_4 \rightarrow A_3, \\ (t=6) & A_3 \rightarrow A_3. \end{array} \tag{7}$$

In (7), three out of six FLRs have the same LHS – A_3. We can pick out those FLRs in the following equation:

$$\begin{array}{ll} (t=3) & A_3 \rightarrow A_3, \\ (t=4) & A_3 \rightarrow A_4, \\ (t=6) & A_3 \rightarrow A_3. \end{array} \tag{8}$$

Following [11], these FLRs in (8) are used to establish such FLR group as

$$A_3 \rightarrow A_3, A_4. \tag{9}$$

When considering the FLRs in (8), instances of identical FLRs are treated as if they were a singular occurrence. Put simply, repeated occurrences of the same FLRs are disregarded in (9).

However, the decision to ignore recurrence raises legitimate concerns. Each occurrence of a specific FLR reflects the frequency of its appearances in the past. For instance, in (8), $A_3 \rightarrow A_3$ appears two times, whereas $A_3 \rightarrow A_4$ appears only once. The recurrence of FLRs can provide valuable insights into their potential future occurrences.

By neglecting the consideration of recurrence, valuable information regarding the frequency and temporal dynamics of FLRs may be overlooked. This omission may hinder the ability of the forecasting model to accurately capture the patterns and trends within the data.

To address this limitation, it is imperative to incorporate the concept of recurrence into the forecasting framework. By acknowledging and utilizing the recurrence of FLRs, the forecasting model can more effectively capture the likelihood and potential impact of future FLR occurrences. This consideration of recurrence empowers the model to make more informed predictions and improve the overall accuracy of the forecasting process.

Subsequently, we present an approach that accounts for the frequency and past occurrences of FLRs. This method aims to enhance the forecasting model's capability to capture and leverage the recurrent patterns within the data. Hence, the weighted method suggests to cover all of the FLRs:

$$A_3 \rightarrow A_3, A_4, A_3. \tag{10}$$

The consideration of recurrence becomes crucial in weighted models when analyzing FLRs. Each FLR's recurrence should be acknowledged and accounted for, leading to the assignment of distinct weights for FLRs with varying recurrences. Determining appropriate weights based on the chronological order of FLRs emerges as a viable approach.

In practice, FLRs from different time slots are often perceived as differing in importance. One possible method involves attributing higher weights to recent FLRs, considering them more significant than older ones. This approach acknowledges the temporal dimension and the potential evolving nature of FLRs, allowing for a more nuanced and contextually relevant interpretation of their impact on the forecasting process.

By assigning weights based on the chronological order of FLRs, the weighted models can better capture the dynamics and relevance of FLRs over time. This approach aligns the forecasting methodology with the natural progression of events, where recent occurrences are more likely to hold greater influence than older ones. Through this weighting scheme, the forecasting model can effectively adapt to changing patterns and prioritize the most pertinent and up-to-date information in making accurate predictions.

In this section, we describe Yu method for the construction of the weighted FLRs. By incorporating this temporal weighting scheme, this method aims to enhance the forecasting model's ability to capture the evolving importance of FLRs and improve the overall accuracy and relevance of the forecasting process.

To apply the chronologically-determined weights, we can simply assign different weights for each FLR incrementally, say 1, 2, and 3:

$$\begin{aligned} &(t=3) \quad A_3 \rightarrow A_3 \quad \text{with weight } 1, \\ &(t=4) \quad A_3 \rightarrow A_4 \quad \text{with weight } 2, \\ &(t=6) \quad A_3 \rightarrow A_3 \quad \text{with weight } 3. \end{aligned} \tag{11}$$

As a result, the most recent FLR $(t = 6)$ is assigned the highest weight of 3, which means that the probability of its appearance in the near future is higher than for previous FLRs. The most aged FLR $(t = 3)$ is assigned the lowest weight of 1, which means that the probability of its appearance in the near future is lower than in the case of the others.

Various weight schemes can be employed for both domain-based and chronologically-determined weightings. It can be emphasized that the models used in previous studies (without explicit weight specifications) can be regarded as instances where equal weights are assigned. Thus, the weighted models represent a broader and more inclusive framework that encompasses these previous models.

To address the limitations of conventional fuzzy time series models, we propose weighted fuzzy time series models as an extension. These weighted models introduce modifications to specific steps within the conventional framework. In particular, Step 5 undergoes revision to address the recurrence issue, ensuring that the impact of repeated patterns is properly considered. Step 6 remains unchanged as it represents the existing defuzzification process. However, Step 7 is modified to accommodate the handling of the defuzzified matrix in light of the weighted approach. Furthermore, the weighted models introduce two additional steps: Step 8, which focuses on the assignment of appropriate weights, and Step 9, which calculates the final forecasting results.

By incorporating these modifications and additions, the weighted fuzzy time series models offer a more comprehensive and refined methodology for capturing and leveraging the inherent dynamics and characteristics of the data. Below, we delve into the details of the weighted fuzzy time series models, elucidating the specific modifications made to the conventional framework and showcasing the benefits and enhanced performance they bring to the forecasting process.

Step 5: Establishing fuzzy relationships (revised).

To account for recurrent FLRs, a revision is made to Step 5 within Chen model. In this context, we refer to the example presented in (7), where multiple fuzzy relations exhibit different linguistic terms on the LHS, specifically A_1 and A_2. As a result, we establish the following FLR groups: $A_1 \to A_2$ and $A_2 \to A_3$.

Additionally, within (7), there are three FLRs featuring the same LHS, namely $A_3 \to A_3$, $A_3 \to A_4$, $A_3 \to A_3$. These FLRs are utilized to form another FLR group, as demonstrated in (10): $A_3 \to A_3, A_4, A_3$.

Step 6: Forecasting.

Given that $F(t-1) = A_1$, the forecast for $F(t)$ will be A_2. This forecast pattern continues for subsequent FLRs: when $F(t-1) = A_2$, the forecast for $F(t)$ will be A_3. However, if $F(t-1) = A_3$, the forecast for $F(t)$ will consist of the fuzzy group $A_3, A_4,\ A_3$.

Step 7: Defuzzification (revised).

When carrying out defuzzification of $F(t)$, we use the matrix of midpoints $M(t) = \left[m_{j1}, m_{j2}, \dots, m_{jg}\right]$ of fuzzy sets' membership functions of RHS of the corresponding FLR group.

Accordingly, for $F(t-1) = A_2$, the midpoint matrix will be defined as $M(t) = [m_3]$, since $F(t) = A_3$. If $F(t) = A_3, A_4, A_3$ in case of $F(t-1) = A_3$, then the corresponding midpoint matrix will be defined as $M(t) = [m_3, m_4, m_3]$.

Step 8: Assigning weights.

Let's assume that the forecast comprises a group of different fuzzy sets $A_{j1}, A_{j2}, \dots, A_{jg}$. To determine their significance for the overall forecast, it is necessary to define weights $w_1, w_2, \dots, w_g$. Before constructing the weight matrix, the following condition must be satisfied:

$$\sum_{h=1}^{g} w'_h = 1. \tag{12}$$

Thus, each weight has to be standardized and the following matrix will be formed:

$$W(t) = [w'_1, w'_2, \dots, w'_g] = \left[\frac{w_1}{\sum_{h=1}^{g} w_h}, \frac{w_2}{\sum_{h=1}^{g} w_h}, \dots, \frac{w_g}{\sum_{h=1}^{g} w_h}\right], \tag{13}$$

where w_h is the weight for each A_{jh}.

For the formation of the weighted matrix, an intuitive scheme of incremental weight coefficients can be proposed. We start with $w_1 = 1$ and

$$w_i = w_{i-1} + 1, \quad i = \overline{2, g}. \tag{14}$$

Hence, the i-th item in (13) can be rewritten as

$$\begin{gathered} \frac{w_i}{\sum_{h=1}^{g} w_h} = \frac{w_{i-1} + 1}{\sum_{h=1}^{g} w_h} = \frac{(w_{i-2} + 1) + 1}{\sum_{h=1}^{g} w_h} = \cdots \\ \cdots = \frac{i}{\sum_{h=1}^{g} w_h} = \frac{i}{\sum_{h=1}^{g} h}. \end{gathered} \tag{15}$$

In this case, (13) becomes

$$\begin{gathered} W(t) = [w'_1, w'_2, \dots, w'_g] = \left[\frac{w_1}{\sum_{h=1}^{g} w_h}, \frac{w_2}{\sum_{h=1}^{g} w_h}, \dots, \frac{w_g}{\sum_{h=1}^{g} w_h}\right] = \\ = \left[\frac{1}{\sum_{h=1}^{g} h}, \frac{2}{\sum_{h=1}^{g} h}, \cdots, \frac{g}{\sum_{h=1}^{g} h}\right]. \end{gathered}$$

Therefore, if the forecast of $F(t)$ is only one term A_2, the weight matrix is determined as $W(t) = \left[\frac{1}{1}\right]$.

In case when the linguistic variable $F(t) = A_3, A_4, A_3$, the weights are specified by following the scheme (14): $w_1 = 1, w_2 = 2, w_3 = 3$. So, the weight matrix is determined as

$$W(t) = \left[\frac{1}{6}, \frac{2}{6}, \frac{3}{6}\right].$$

Step 9: Calculating results.

The final forecast is calculated as the product of the defuzzified matrix and the transpose of the weight matrix:

$$\hat{y}(t) = M(t) \times W(t)^T. \tag{16}$$

According to our examples, for $F(t) = A_3$

$$\hat{y}(t) = [m_3] \times \left[\frac{1}{1}\right]^T, \tag{17}$$

and, for $F(t) = A_3, A_4, A_3$

$$\hat{y}(t) = [m_3, m_4, m_3] \times \left[\frac{1}{6}, \frac{2}{6}, \frac{3}{6}\right]^T. \tag{18}$$

Empirical analysis and forecasting

In this section we test the effectiveness and robustness of the described fuzzy models to yield accurate enough forecasting results. Here, we propose to test the effectiveness of these methods for the daily closing prices data of Bitcoin (BTC-USD) for the period from January 1, 2016, to September 14, 2022. Data was obtained from Yahoo! Finance [58].

As stated previously, data preprocessing is one of the key components for further partitioning, fuzzification, and knowledge extraction. Since cryptocurrency data is presented by a time-varying time series, it is important to stabilize its dynamics, i.e., make it stationary. Relying on [46], we use Return on Investment (ROI), which is the measure of efficiency of an investment:

$$ROI = \frac{Gains - Investment\ Cost}{Investment\ Costs}. \quad (19)$$

For a time series Y, we define daily ROI as follows:

$$ROI(t) = \frac{y(t) - y(t-1)}{y(t-1)}. \quad (20)$$

Applying (20), we work in new domain with less noisy and volatile effects.

In our case, data is preprocessed, then initial forecast can be obtained by the following reversing operation:

$$\hat{y}(t) = y(t-1)\left(\widehat{ROI}(t) + 1\right). \quad (21)$$

According to (21), $\hat{y}(t)$ is forecast of the base indicator at time t and $\widehat{ROI}(t)$ is the forecasted returns at time t.

Initial BTC-USD time series and its returns are presented in Figs. 2 and 3.

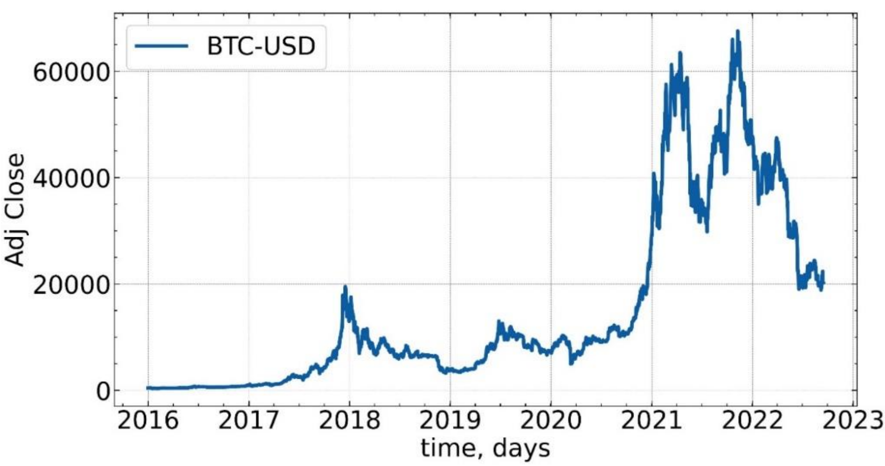

Fig. 2. Initial dynamics of adjusted closing prices of BTC-USD

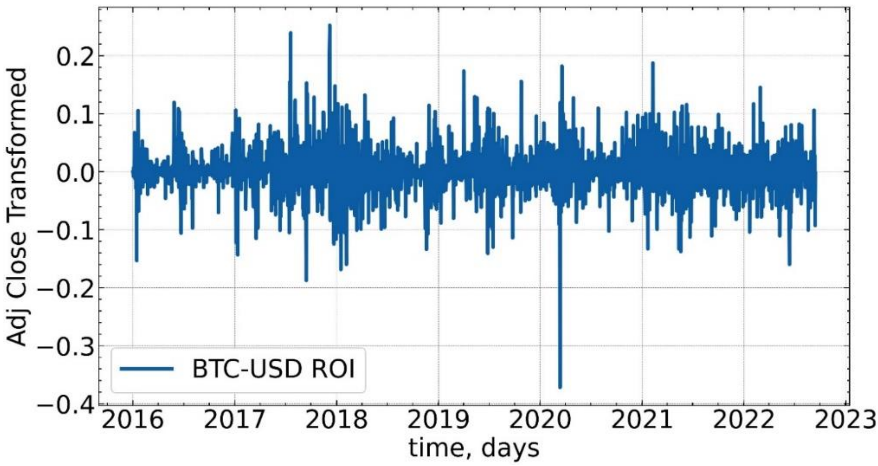

Fig. 3. BTC-USD adjusted closing prices returns

For further forecasting procedure, we split initial time series into training and testing sets. Training set is used to define linguistic variable, its fuzzy sets and membership functions and, based on them, obtain an optimal group of fuzzy logical rules. For the training dataset, we select 80% of the values from the entire sample. The remaining 20% are included in the test set. The test set of data will be used to verify the adequacy of the obtained fuzzy logic rules used in the algorithms described above. Since the test dataset is a sample that our algorithms did not see during training, so the metrics of model quality on this set will have to tell how suitable these approaches will be for further usage.

Next, we divide our universe of discourse into partitions of equal length, on which we define fuzzy sets. In this study, according to our estimations, we have split BTC-USD dynamics into 15 fuzzy sets using Gaussian membership function, which are presented in Fig. 4 and Table 2. Therefore, we have obtained linguistic variable "BTC-USD fuzzy time series" with 15 linguistic terms corresponding to their fuzzy sets. On their basis, we determine fuzzy logical relationships and group them.

In the case of choosing the Gaussian membership function, each fuzzy set is determined by its mean m and variance σ, which are presented in Table 2.

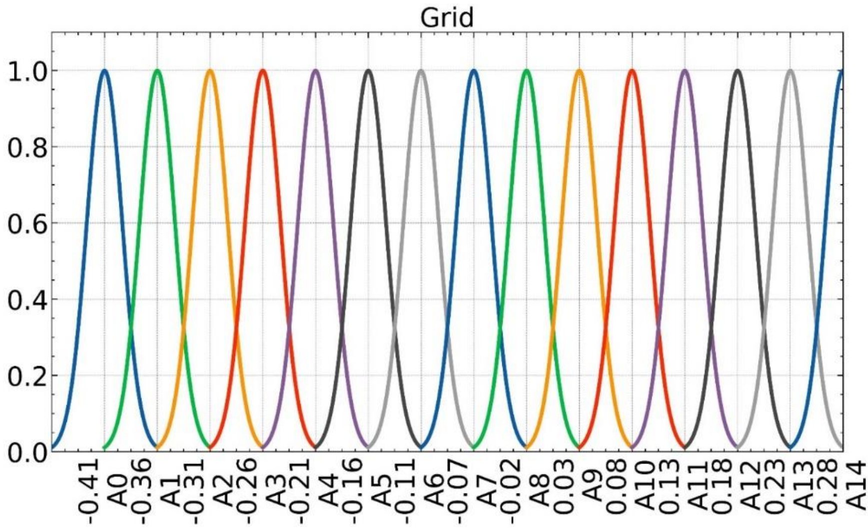

Fig. 4. UoD Partitioning using Gaussian membership function for BTC time series

Table 2

DESCRIPTIVE STATISTICS OF EACH FUZZY SET

Fuzzy set	m	σ
A0	-0.41	0.02
A1	-0.36	
A2	-0.31	
A3	-0.26	
A4	-0.21	
A5	-0.16	
A6	-0.11	
A7	-0.07	
A8	-0.02	
A9	0.03	
A10	0.08	
A11	0.13	
A12	0.18	
A13	0.23	
A14	0.28	

The result of grouping fuzzy logical relationships shown in Table 3.

Table 3

FLRS GROUPING BY THE EXAMPLE OF CHEN MODEL

LHS	RHS
A13	A7
A11	A6, A7, A8, A9
A10	A10, A5, A6, A7, A8, A9
A6	A10, A5, A6, A7, A8, A9
A7	A10, A5, A6, A7, A8, A9
A8	A1, A10, A11, A12, A13, A5, A6, A7, A8, A9
A9	A10, A11, A12, A6, A7, A8, A9
A5	A10, A11, A8, A9
A12	A10, A13, A8, A9
A1	A11

By applying the described models, the forecasted closing prices of BTC-USD from 2021 to 2022 can be evaluated. Fig. 5 shows the results of three fuzzy prediction models for the test set of BTC-USD closing prices.

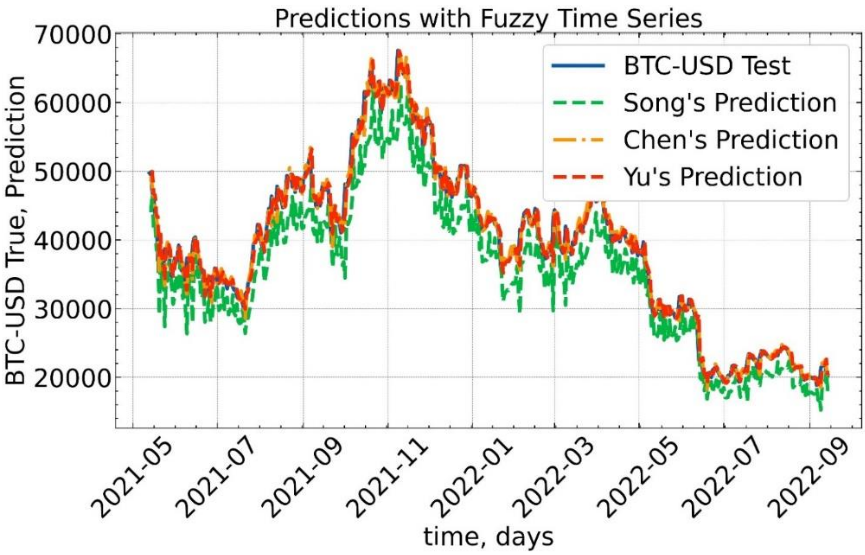

Fig. 5. One step ahead forecasting for a test set of BTC-USD time series using the Song, Chen, and Yu models

The empirical results show that the forecast values of the three models correlate quite closely with the true dynamics of the studied cryptocurrency. The predictions for the Song model look a bit shifted by a specific value. This problem could be solved by choosing, for example, a different representation of the series rather than returns. Nevertheless, all three models capture the trend component of BTC-USD quite accurately.

It is also important to perform a residual analysis in order to see if their properties fit into the white noise assumption, such that $\varepsilon \sim \mathcal{N}(0,1)$. Several statistical tests have been proposed in the literature to verify this condition. One such test is the Box-Pierce test [9]. Another improved version of this test is the Ljung-Box test [37].

The Ljung-Box test examines the hypotheses H_0 and H_1 for each lag of the autocorrelation function of the residuals. H_0 assumes that the residuals are independent and identically distributed (i.i.d), while H_1 suggests that they are not i.i.d. To reject H_0, the test statistic Q must exceed the critical value $\mathcal{X}^2_{1-\alpha,df}$, where α represents the confidence level and df corresponds to the lag number.

The Augmented Dickey–Fuller test (ADF) is among the most frequently employed tests for examining the existence of a unit root in a time series [19]. A unit root represents a particular form of non-stationarity, where the time series exhibits a persistent stochastic trend. A time series with a unit root is referred to as being integrated of order d, denoted as $I(d)$, if it becomes stationary after undergoing differencing d times. For instance, if a time series is $I(1)$, it signifies that it is non-stationary, but its first difference is stationary. The ADF test is based on the concept of assessing whether the coefficient of the lagged level of the time series is equal to one in a regression equation. The null hypothesis of the test posits that the time series possesses a unit root, while the alternative hypothesis suggests stationarity.

In order to conduct the ADF test, it is necessary to select the suitable configuration of the regression equation and the number of lagged differences to incorporate. The choice of specification is contingent upon whether one anticipates the time series to possess a constant mean, a linear trend, or neither. The determination of the number of lags can be accomplished by employing information criteria, such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC), which strike a balance between model fit and complexity. In our case, we use constant mean in regression model and AIC for the ADF test.

Upon conducting the ADF test, we obtain a test statistic and a corresponding p-values for each model. The test statistic is then compared to critical values at various significance levels, typically 1%, 5%, and 10%. If the test statistic is more negative than the critical value, it is possible to reject the null hypothesis and infer that the time series exhibits stationarity. On the other hand, if the test statistic is less negative than the critical value, it is not possible to reject the null hypothesis, thereby suggesting the presence of a unit root in the time series. Also p-value is compared to the critical value $\alpha = 0.05$. Therefore, if the p-value will be less than the level α, we can conclude that our residuals are stationary.

In Figs. 6-8 are presented the following illustrations for all built models: model residuals together with the moving average and standard deviation calculated in a window of 12 values; autocorrelation and partial autocorrelation function; Q-Q plot; histogram of model residuals. Below each figure are tables with statistics for each test.

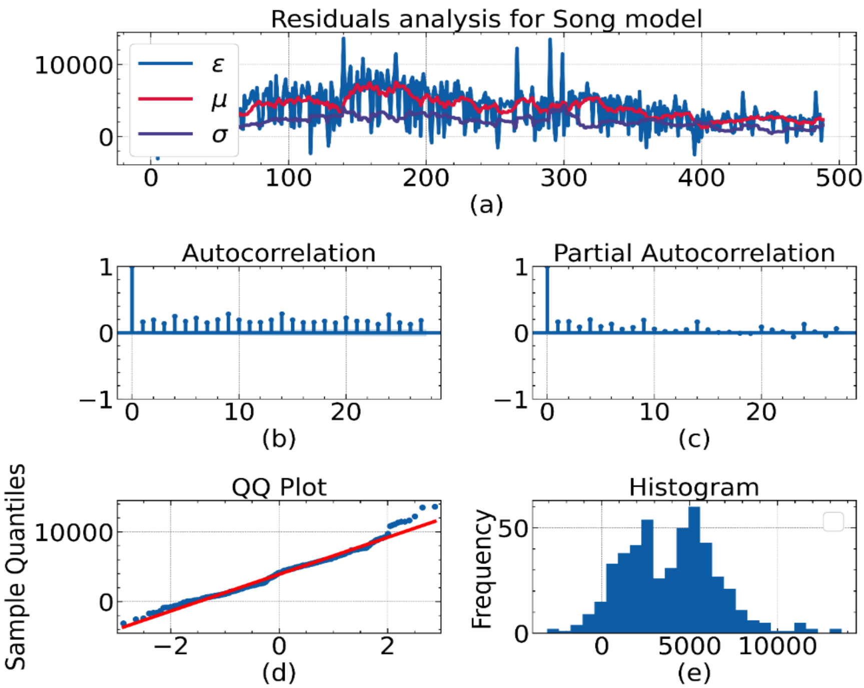

Fig. 6. Analysis of prediction residuals by Song model:
residuals ε, moving average μ, and standard deviation σ (a), autocorrelation functions (b) and (c), Q-Q plot (d), and residuals histogram (e)

As the results of calculations show, Song model is not optimal enough to make forecasts on the BTC-USD market. Firstly, visually the residuals are shifted (see Fig. 6.a), that is, above zero, although ideally they should be at zero most of the time. Based on the autocorrelation graphs (Figs. 6.b, 6.c), we can say that the residuals contain information of a systematic nature that was not taken into account by the model. The Q-Q graph shows (see Fig. 6.d) that we have a large outlier in the right tail. According to the histogram of the residuals at Fig. 6.e, the offset from zero is more demonstrative.

The test results presented in Tables 4 and 5 should also not be reassuring.

Table 4

RESULTS OF AUGMENTED DICKEY-FULLER TEST FOR SONG MODEL

Statistic characteristics	**Value**
Test Statistic	-2.15
p-value	0.23
Critical Value (1%)	-3.44
Critical Value (5%)	-2.87
Critical Value (10%)	-2.57

Table 5

LJUNG-BOX TEST FOR SONG MODEL RESIDUALS

Lag	**Test Statistic**	$\boldsymbol{p}$**-value**
1	13.38	$2.54 \cdot 10^{-4}$
2	32.00	$1.13 \cdot 10^{-7}$
3	41.47	$5.20 \cdot 10^{-9}$
4	71.99	$8.61 \cdot 10^{-15}$
5	87.15	$2.66 \cdot 10^{-17}$

According to the values obtained in accordance with the ADF test, it is clear that our residuals are not stationary, since the p-value in this case is greater than 0.05. The Ljung-Box test conducted for the first 5 lags shows that the p-values are less than 0.05. This suggests that there is a significant autocorrelation in the residuals of the model.

We perform the same analysis of the residuals for the Chen model. Graphical illustrations are presented in Fig. 7.

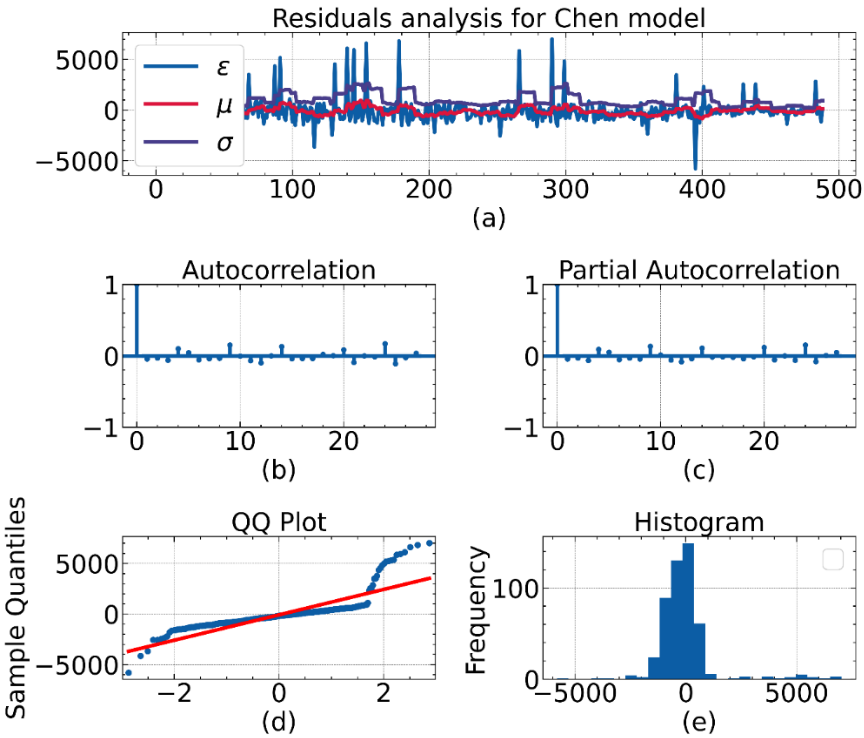

Fig. 7. Analysis of prediction residuals by Chen model: residuals ε, moving average μ, and standard deviation σ (a), autocorrelation functions (b) and (c), Q-Q plot (d), and residuals histogram (e)

Based on the dynamics of the residuals in Chen model in Fig. 7.a, it can be seen that although they have outliers, they are not as biased as those in Song model. Minor outliers are observed in ordinary and partial autocorrelation, however, they are mostly at zero for the selected 30 lags (see Figs. 7.b, 7.c). The Q-Q graph in Fig. 7.d shows that our forecasts have outliers that we do not take into account well. However, for most values, our model does well. The histogram of the residuals is much better concentrated in the zero area when compared with the previous results (see Figs. 6.e, 7.e). And it can be seen that it is much more similar to a normal Gaussian distribution.

Now let us look at the results in Tables 6 and 7 according to the ADF and Ljung-Box tests.

Table 6

RESULTS OF AUGMENTED DICKEY-FULLER TEST FOR CHEN MODEL

Statistic characteristics	Value
Test statistic	-5.61
p-value	0.000001
Critical Value (1%)	-3.44
Critical Value (5%)	-2.87
Critical Value (10%)	-2.57

Table 7

LJUNG-BOX TEST FOR CHEN MODEL RESIDUALS

Lag	Test Statistic	p-value
1	0.96	0.33
2	1.40	0.49
3	3.35	0.34
4	8.18	0.09
5	9.21	0.10

The results of the ADF test show that the p-value is less than 0.05, which indicates that our residuals are stationary. The Ljung-Box test for the first five lags of the autocorrelation function indicates towards the null hypothesis. That is, the residuals are independent and identically distributed.

It remains to analyze the residuals in the Yu model. The results of modeling are presented in Fig. 8.

Similarly to the results in Chen model, Yu model residuals are less biased compared to those observed in Song model. Although minor outliers are present in the standard and partial autocorrelation plots, they are predominantly centered around zero across the selected 30 lags, as can be seen in Figs. 8.b and 8.c. The Q-Q graph in Fig. 8.d indicates that our forecasts exhibit outliers that are not adequately accounted for. Nevertheless, our model performs satisfactorily for the majority of values. Furthermore, the histogram of the residuals at Fig. 8.e demonstrates a significant concentration around zero, surpassing the previous results. Additionally, it exhibits a closer resemblance to a normal Gaussian distribution.

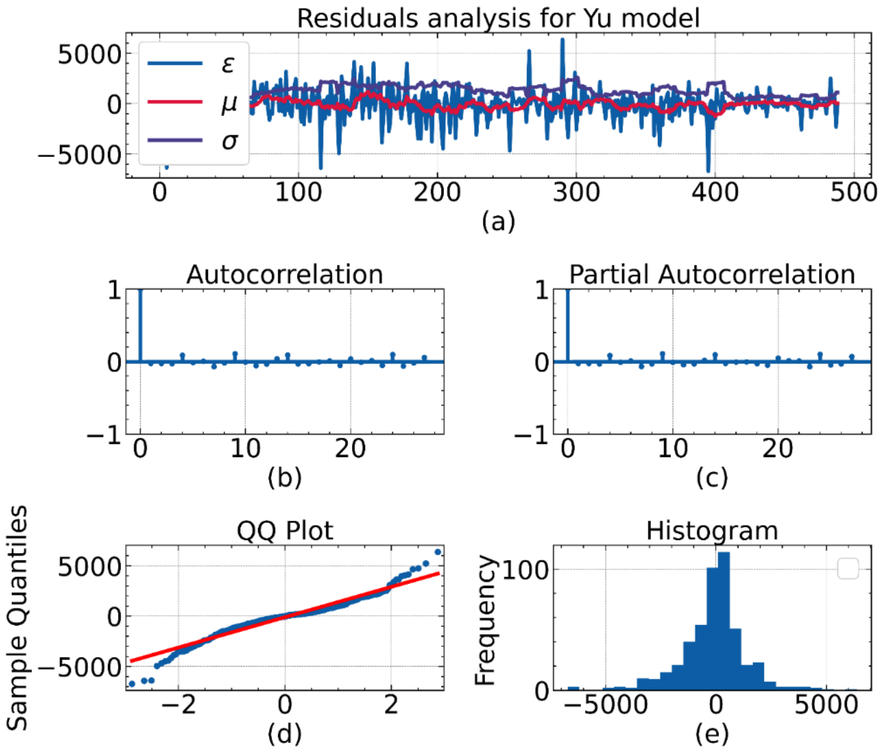

Fig. 8. Analysis of prediction residuals by Yu model:
residuals ε, moving average μ, and standard deviation σ (a), autocorrelation functions (b) and (c), Q-Q plot (d), and residuals histogram (e)

Next, let us present quantitative estimates of the forecasting results in Tables 8 and 9.

Table 8

RESULTS OF AUGMENTED DICKEY-FULLER TEST FOR YU MODEL

Statistic characteristics	Value
Test statistic	-16.16
p-value	$4.45 \cdot 10^{-29}$
Critical Value (1%)	-3.44
Critical Value (5%)	-2.87
Critical Value (10%)	-2.57

Table 9

LJUNG-BOX TEST FOR YU MODEL RESIDUALS

Lag	Test statistic	p-value
1	0.28	0.60
2	0.55	0.76
3	0.97	0.81
4	4.73	0.32
5	4.82	0.44

The outcomes of the ADF test reveal a p-value below 0.05 (as can be seen in Table 8), indicating that our residuals are stationary. Similarly, the Ljung-Box test conducted on the initial five lags of the autocorrelation function supports the null hypothesis, suggesting that the residuals maintain independence and identical distribution.

After forecasting and residuals analysis, different forecasting metrics were obtained for testing the described models. Here we present the following metrics:

- Mean Absolute Error (MAE), which is calculated as mean of absolute deviations of predictions $\hat{y}(t)$ from true values $y(t)$:

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|y(t) - \hat{y}(t)|, \tag{22}$$

where N is a number of the test sample elements.

- Root Mean Squared Error (RMSE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\big(y(t) - \hat{y}(t)\big)^2}. \tag{23}$$

- Coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum_{t=1}^{N}\big(y(t) - \hat{y}(t)\big)^2}{\sum_{t=1}^{N}(y(t) - \bar{y})^2}, \tag{24}$$

where $\bar{y}$ is the average of Y on the test sample.

- Accuracy:

$$Accuracy = 100 - \left(\frac{100}{N} \sum_{t=1}^{N} \frac{|y(t) - \hat{y}(t)|}{y(t)} \right). \quad (25)$$

Tables 10-12 show values of the accuracy metrics according to (22)-(25).

Table 10

FORECASTING METRICS FOR SONG MODEL

Mean Absolute Error	4020.13
Root Mean Squared Error	4738.78
Coefficient of determination	0.84
Accuracy	89.78 %

Table 11

FORECASTING METRICS FOR CHEN MODEL

Mean Absolute Error	711.92
Root Mean Squared Error	1259.81
Coefficient of determination	0.99
Accuracy	98.14 %

Table 12

FORECASTING METRICS FOR YU MODEL

Mean Absolute Error	1027.96
Root Mean Squared Error	1509.55
Coefficient of determination	0.98
Accuracy	97.32 %

Considering the estimates of all metrics on the mentioned tables, and relying on the analysis of the residuals, we can say that the Chen and Yu models seem more suitable for the task of forecasting the cryptocurrency market. For example, the average forecast error on the test set for the Chen model is $712, and for the Yu model it is about $1028. As can be seen for the Song model, the mean absolute error is large, although the coefficient of determination and accuracy appear to be quite high. Nevertheless, remembering the testing of residuals, it cannot be said that Song model is sufficiently reliable. According to the coefficient of determination and accuracy of the last two models, one can think about the sufficiently high prospects of their further use for building trading strategies.

Next, let's try to look at the accuracy of the forecasts of the presented models for each year. Here we will focus on the absolute percentage error (APE). The box diagram should show us how close our forecasts are on average to real values, and how large our errors are. The APE metric is calculated as follows:

$$APE = \frac{100}{N} \cdot \sum_{t=1}^{N} \frac{|y(t) - \hat{y}(t)|}{y(t)}. \tag{26}$$

Figs. 9-11 present APE calculated for the forecasts of each model and within each year of testing set.

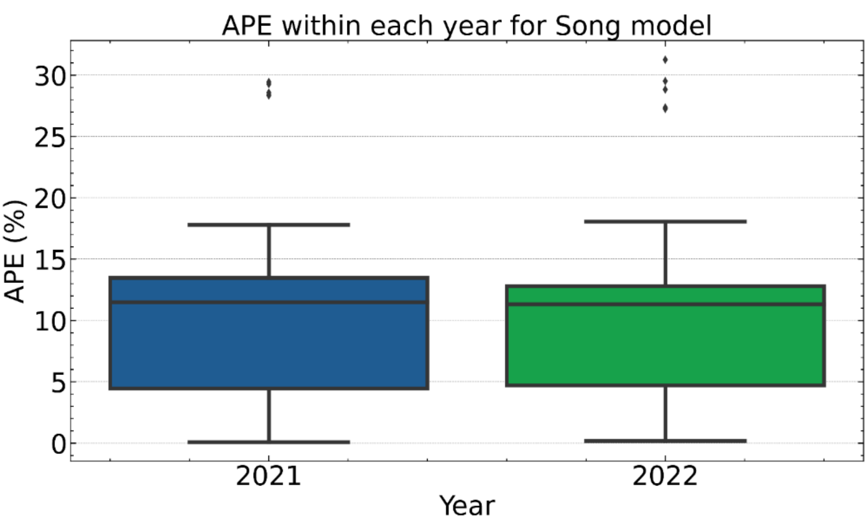

Fig. 9. Box plots of APE within each year of BTC-USD test set obtained from Song model

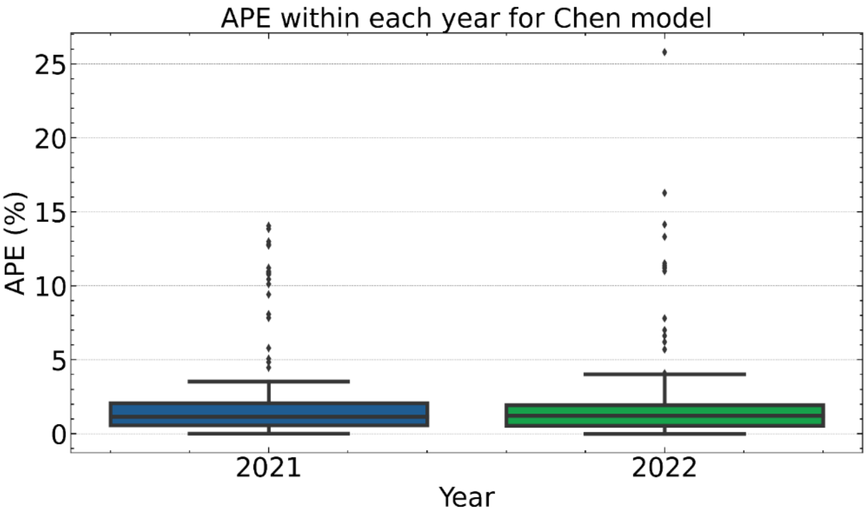

Fig. 10. Box plots of APE within each year of BTC-USD test set obtained from Chen model

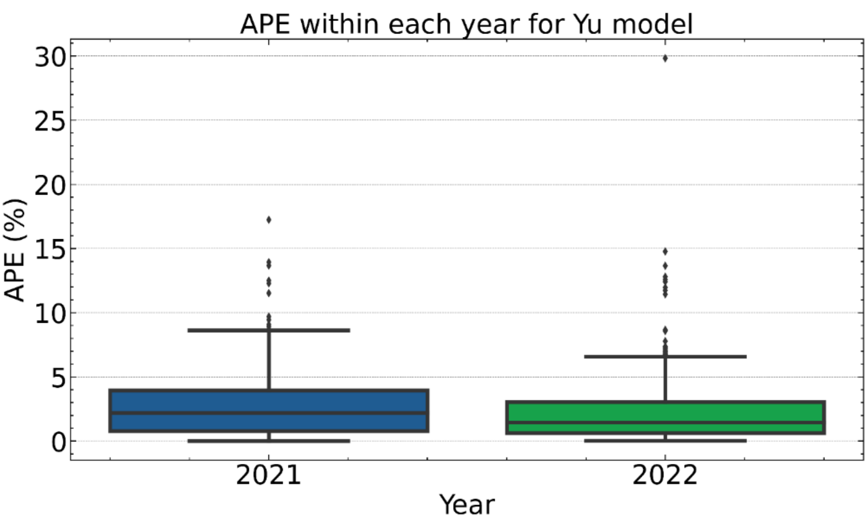

Fig. 11. Box plots of APE within each year of BTC-USD test set obtained from Yu model

Considering the box diagrams of the absolute percentage errors of the studied models within one year, it can be seen that the forecasts for Chen and Yu models are more accurate. For Song model, the absolute percentage error is around 10-15%, but for the last two models it remains around zero. All of the presented models are characterized by significant outliers, which are close to almost 30%. For the Chen and Yu models, the median value of APE is close to zero. Therefore, these models are not perfect, and even in their forecasts there are significant errors due to their specification.

Conclusions

In recent times, Fuzzy Time Series methods have emerged as an increasingly prominent area of research that has garnered growing interest. These FTS forecasting approaches yield models that are driven by data and do not rely on explicit parameters, making them appealing due to their simplicity, adaptability, forecasting precision, and computational efficiency. Additionally, FTS methods offer the advantage of generating interpretable representations of time series patterns, enabling the transfer, auditability, reusability, and upgradability of knowledge derived from these models. In the context of this study, we specifically investigate various Fuzzy Time Series methods as applied to the BTC-USD market. Among these variants, our research focuses on time-invariant rule-based FTS techniques. Specifically, we examined the conventional First-Order Fuzzy Time Series (Song and Chen) models and the Weighted First-Order Fuzzy Time Series (Yu) model. We rigorously assess and compare the accuracy of these methods, revealing an equivalence in their predictive performance.

Furthermore, in this study, we aim to expand the understanding and application of FTS methods in the realm of cryptocurrency forecasting. By delving into the intricacies of First-Order FTS and Weighted First-Order FTS, we uncover valuable insights into their strengths, limitations, and potential areas for their improvement. Through comprehensive experimental evaluations and rigorous statistical analyses, we shed light on the capabilities of these FTS models in capturing the dynamic and volatile nature of the Bitcoin market. By elucidating the comparative accuracy and computational efficiency of these models, we contributed to the existing body of knowledge and offered practical recommendations for researchers and practitioners seeking reliable forecasting methodologies in the cryptocurrency domain.

In the future, it would be interesting to consider multivariate fuzzy time series models along with some indicators-precursors based on complex systems theory, and machine (deep) learning methods.

Acknowledgements

This work is a part of the applied research “Monitoring, Forecasting, and Prevention of Crisis Phenomena in Complex Socio-Economic Systems”, which is funded by the Ministry of Education and Science of Ukraine (project No. 0122U001694).

References

1. Bajestani, N. S., & Zare, A. (2011). Forecasting TAIEX using improved type 2 fuzzy time series. *Expert Systems with Applications*, *38*(5), 5816–5821. https://doi.org/10.1016/j.eswa.2010.10.049
2. Bielinskyi, A., & Soloviev, V. (2018). Complex network precursors of crashes and critical events in the cryptocurrency market. *CEUR Workshop Proceeding*, *2292*, 37-45. https://ceur-ws.org/Vol-2292/paper02.pdf
3. Bielinskyi, A., Hushko, S., Matviychuk, A., Serdyuk, O., Semerikov, S., & Soloviev, V. (2021). Irreversibility of financial time series: a case of crisis. *CEUR Workshop Proceedings*, *3048*, 134-150. https://ceur-ws.org/Vol-3048/paper04.pdf
4. Bielinskyi, A., Serdyuk, O., Semerikov, S., & Soloviev, V. (2021). Econophysics of cryptocurrency crashes: a systematic review. *CEUR Workshop Proceedings*, *3048*, 31-133. https://ceur-ws.org/Vol-3048/paper03.pdf
5. Bielinskyi, A., Soloviev, V., Semerikov, S., & Solovieva, V. (2021). Identifying stock market crashes by fuzzy measures of complexity. *Neuro-Fuzzy Modeling Techniques in Economics*, *10*, 3-45. http://doi.org/10.33111/nfmte.2021.003
6. Bielinskyi, A.O., Matviychuk, A.V., Serdyuk, O.A., Semerikov, S.O., Solovieva, V.V., & Soloviev, V.N. (2022). Correlational and Non-extensive

Nature of Carbon Dioxide Pricing Market. In O. Ignatenko, V. Kharchenko, V. Kobets, et al. (Eds.). *Communications in Computer and Information Science: Vol. 1635. ICTERI 2021 Workshops* (pp. 183–199). Springer, Cham. https://doi.org/10.1007/978-3-031-14841-5_12

7. Bitencourt, H. V., & Guimarães, F. G. (2021). High-dimensional Multivariate Time Series Forecasting in IoT Applications using Embedding Non-stationary Fuzzy Time Series. In *Proceedings of 2021 IEEE Latin American Conference on Computational Intelligence* (pp. 1-6). IEEE. https://doi.org/10.1109/la-cci48322.2021.9769792

8. Bose, M., & Mali, K. (2018). A novel data partitioning and rule selection technique for modeling high-order fuzzy time series. *Applied Soft Computing*, *63*, 87–96. https://doi.org/10.1016/j.asoc.2017.11.011

9. Box, G. E. P., & Pierce, D. A. (1970). Distribution of Residual Autocorrelations in Autoregressive-Integrated Moving Average Time Series Models. *Journal of the American Statistical Association*, *65*(332), 1509-1526. https://doi.org/10.2307/2284333

10. Chen, M. Y. (2014). A high-order fuzzy time series forecasting model for internet stock trading. *Future Generation Computer Systems*, *37*, 461–467. https://doi.org/10.1016/j.future.2013.09.025

11. Chen, S. (1996). Forecasting enrollments based on fuzzy time series. *Fuzzy Sets and Systems*, *81*(3), 311–319. https://doi.org/10.1016/0165-0114(95)00220-0

12. Chen, S., & Chang, Y. (2010). Multi-variable fuzzy forecasting based on fuzzy clustering and fuzzy rule interpolation techniques. *Information Sciences*, *180*(24), 4772–4783. https://doi.org/10.1016/j.ins.2010.08.026

13. De Lima E Silva, P. C., Severiano, C. A., Alves, M. A., Silva, R., Cohen, M. W., & Guimarães, F. G. (2020). Forecasting in non-stationary environments with fuzzy time series. *Applied Soft Computing*, *97*(B), Article 106825. https://doi.org/10.1016/j.asoc.2020.106825

14. De Lima E Silva, P.C. (2019). *Scalable Models For Probabilistic Forecasting With Fuzzy Time Series* [Doctoral dissertation, Universidade Federal de Minas Gerais]. Zenodo. https://doi.org/10.5281/zenodo.3449195

15. De Oliveira E Lucas, P., Orang, O., De Lima E Silva, P. C., Mendes, E., & Guimarães, F. G. (2021). A Tutorial on Fuzzy Time Series Forecasting Models: Recent Advances and Challenges. *Learning and Nonlinear Models*, *19*(2), 29–50. https://doi.org/10.21528/lnlm-vol19-no2-art3

16. Derbentsev, V., Matviychuk, A., & Soloviev, V.N. (2020). Forecasting of Cryptocurrency Prices Using Machine Learning. In L. Pichl, C. Eom, E. Scalas, & T. Kaizoji (Eds.), *Advanced Studies of Financial Technologies and Cryptocurrency Markets* (pp. 211-231). Springer, Singapore. https://doi.org/10.1007/978-981-15-4498-9_12

17. Derbentsev, V., Velykoivanenko, H., & Datsenko, N. (2019). Machine learning approach for forecasting cryptocurrencies time series. *Neuro-Fuzzy Modeling Techniques in Economics, 8*, 65-93. http://doi.org/10.33111/nfmte.2019.065

18. Derbentsev, V., Bezkorovainyi, V., Silchenko, M., Hrabariev, A., & Pomazun, O. (2021). Deep Learning Approach for Short-Term Forecasting Trend Movement of Stock Indeces. In *2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T)* (pp. 607-612). IEEE. https://doi.org/10.1109/PICST54195.2021.9772235

19. Dickey, D. A., & Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. *Journal of the American Statistical Association*, *74*(366), 427–431. https://doi.org/10.2307/2286348

20. Dong, Q., & Ma, X. (2021). Enhanced fuzzy time series forecasting model based on hesitant differential fuzzy sets and error learning. *Expert Systems with Applications*, *166*, Article 114056. https://doi.org/10.1016/j.eswa.2020.114056

21. Dos Santos, M. M., Guimarães, F. G., & De Lima E Silva, P. C. (2021). High-dimensional Multivariate Time Series Forecasting using Self-Organizing Maps and Fuzzy Time Series. In *Proceedings of 2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (pp. 1-6). IEEE. https://doi.org/10.1109/fuzz45933.2021.9494496

22. Dubois, D., & Prade, H. (1991). Fuzzy sets in approximate reasoning, Part 1: Inference with possibility distributions. *Fuzzy Sets and Systems*, *40*(1), 143–202. https://doi.org/10.1016/0165-0114(91)90050-z

23. Egrioglu, E., Aladag, C. H., Yolcu, U., Basaran, M. A., & Uslu, V. R. (2009). A new hybrid approach based on SARIMA and partial high order bivariate fuzzy time series forecasting model. *Expert Systems with Applications*, *36*(4), 7424–7434. https://doi.org/10.1016/j.eswa.2008.09.040

24. Egrioglu, E., Aladag, C. H., Yolcu, U., Uslu, V. R., & Basaran, M. A. (2009). A new approach based on artificial neural networks for high order multivariate fuzzy time series. *Expert Systems with Applications*, *36*(7), 10589–10594. https://doi.org/10.1016/j.eswa.2009.02.057

25. Egrioglu, E., Bas, E., Yolcu, U., & Chen, M. Y. (2020). Picture fuzzy time series: Defining, modeling and creating a new forecasting method. *Engineering Applications of Artificial Intelligence*, *88*, Article 103367. https://doi.org/10.1016/j.engappai.2019.103367

26. Huarng, K. (2001). Heuristic models of fuzzy time series for forecasting. *Fuzzy Sets and Systems*, *123*(3), 369–386. https://doi.org/10.1016/s0165-0114(00)00093-2

27. Hwang, J.-R., Chen, S.-M., & Lee, C.-H. (1998). Handling forecasting problems using fuzzy time series. *Fuzzy Sets and Systems*, *100*(1–3), 217–228. https://doi.org/10.1016/s0165-0114(97)00121-8

28. Ivanchenko, H., & Vashchaiev, S. (2018). Studying the dynamics of nonlinear interaction between enterprise populations. *Neuro-Fuzzy Modeling Techniques in Economics*, *7*, 78-110. http://doi.org/10.33111/nfmte.2018.078

29. Jin, X., Wang, Y.-X., & Yan, X. (2021). Inter-Series Attention Model for COVID-19 Forecasting. In C. Demeniconi, I. Davidson, L. Akoglu, & E. Terzi (Eds.), *Proceedings of the 2021 SIAM International Conference on Data Mining* (pp. 495–503). SIAM. https://doi.org/10.1137/1.9781611976700.56

30. Kaminskyi, A., Miroshnychenko, I., & Pysanets, K. (2019). Risk and return for cryptocurrencies as alternative investment: Kohonen maps clustering. *Neuro-Fuzzy Modeling Techniques in Economics*, *8*, 175-193. http://doi.org/10.33111/nfmte.2019.175

31. Kmytiuk, T., & Majore, G. (2021). Time series forecasting of agricultural product prices using Elman and Jordan recurrent neural networks. *Neuro-Fuzzy Modeling Techniques in Economics*, *10*, 67-85. http://doi.org/10.33111/nfmte.2021.067

32. Kobets, V., & Novak, O. (2021). EU countries clustering for the state of food security using machine learning techniques. *Neuro-Fuzzy Modeling Techniques in Economics*, *10*, 86-118. http://doi.org/10.33111/nfmte.2021.086

33. Kozlovskyi, S., Mazur, H., Vdovenko, N., Shepel, T., & Kozlovskyi, V. (2018). Modeling and forecasting the level of state stimulation of agricultural production in Ukraine based on the theory of fuzzy logic. *Montenegrin Journal of Economics*, *14*(3), 37-53. https://doi.org/10.14254/1800-5845/2018.14-3.3

34. Lara-Benítez, P., Carranza-García, M., & Riquelme, J. C. (2021). An Experimental Review on Deep Learning Architectures for Time Series Forecasting. *International Journal of Neural Systems*, *31*(03), Article 2130001. https://doi.org/10.1142/s0129065721300011

35. Lee, C.-H. L., Liu, A., & Chen, W.-S. (2006). Pattern discovery of fuzzy time series for financial prediction. *IEEE Transactions on Knowledge and Data Engineering*, *18*(5), 613–625. https://doi.org/10.1109/tkde.2006.80

36. Liu, Y., Wu, H., Wang, J., & Long, M. (2022). *Non-stationary transformers: Rethinking the stationarity in time series forecasting*. arXiv. https://arxiv.org/abs/2205.14415

37. Ljung, G. M., & Box, G. E. P. (1978). On a measure of lack of fit in time series models. *Biometrika*, *65*(2), 297–303. https://doi.org/10.1093/biomet/65.2.297

38. Luo, C. B., & Wang, H. (2020). Fuzzy forecasting for long-term time series based on time-variant fuzzy information granules. *Applied Soft Computing*, *88*, Article 106046. https://doi.org/10.1016/j.asoc.2019.106046

39. Matviychuk, A. (2006). Fuzzy logic approach to identification and forecasting of financial time series using Elliott wave theory. *Fuzzy economic review, 11*(2), 51-68. https://doi.org/10.25102/fer.2006.02.04

40. Matviychuk, A., Lukianenko, O., & Miroshnychenko, I. (2019). Neuro-fuzzy model of country's investment potential assessment. *Fuzzy economic review, 24*(2), 65-88. https://doi.org/10.25102/fer.2019.02.04

41. Matviychuk, A., Strelchenko, I., Vashchaiev, S., & Velykoivanenko, H. (2019). Simulation of the Crisis Contagion Process Between Countries with Different Levels of Socio-Economic Development. *CEUR Workshop Proceedings*, *2393*, 485-496. http://ceur-ws.org/Vol-2393/paper_423.pdf

42. Olivares, K. G., Challu, C., Marcjasz, G., Weron, R., & Dubrawski, A. (2021). *Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with NBEATSx*. arXiv. https://arxiv.org/abs/2104.05522v6

43. Pinto, A.C.V., Fernandes, T.E., De Lima E Silva, P.C.L., Guimarães, F.G., Wagner, C., & De Aguiar, E. P. (2022). Interval type-2 fuzzy set based time series forecasting using a data-driven partitioning approach. *Evolving Systems*, *13*, 703–721. https://doi.org/10.1007/s12530-022-09452-2

44. Rangapuram, S. S., Werner, L., Benidis, K., Mercado, P., Gasthaus, J., & Januschowski, T. (2021). End-to-End Learning of Coherent Probabilistic Forecasts for Hierarchical Time Series. In M. Meila, & T. Zhang (Eds.), *Proceedings of Machine Learning Research: Vol. 139. Proceedings of the 38th International Conference on Machine Learning* (pp. 8832–8843). PMLR. http://proceedings.mlr.press/v139/rangapuram21a/rangapuram21a.pdf

45. Rasul, K., Seward, C., Schuster, I., & Vollgraf, R. (2021). Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting. In M. Meila, & T. Zhang (Eds.), *Proceedings of Machine Learning Research: Vol. 139. Proceedings of the 38th International Conference on Machine Learning* (pp. 8857–8868). PMLR. http://proceedings.mlr.press/v139/rasul21a/rasul21a.pdf

46. Sadaei, H. J., & Lee, M. H. (2014). Multilayer Stock Forecasting Model Using Fuzzy Time Series. *The Scientific World Journal*, *2014*, 1–10. https://doi.org/10.1155/2014/610594

47. Shah, M. (2012). Fuzzy based trend mapping and forecasting for time series data. *Expert Systems with Applications*, *39*(7), 6351-6358. https://doi.org/10.1016/j.eswa.2011.12.036

48. Singh, P., & Borah, B. (2013). High-order fuzzy-neuro expert system for time series forecasting. *Knowledge Based Systems*, *46*, 12–21. https://doi.org/10.1016/j.knosys.2013.01.030

49. Soloviev, V., Bielinskyi, A., & Solovieva, V. (2019). Entropy analysis of crisis phenomena for DJIA index. *CEUR Workshop Proceedings, 2393*, 434-449. http://ceur-ws.org/Vol-2393/paper_375.pdf

50. Soloviev, V., Bielinskyi, A., Serdyuk, O., Solovieva, V., & Semerikov, S. (2020). Lyapunov exponents as indicators of the stock market crashes. *CEUR Workshop Proceedings*, *2732*, 455-470. http://ceur-ws.org/Vol-2732/20200455.pdf

51. Soloviev, V., Solovieva, V., & Tuliakova, A. (2019). Visibility graphs and precursors of stock crashes. *Neuro-Fuzzy Modeling Techniques in Economics*, *8*, 3-29. http://doi.org/10.33111/nfmte.2019.003

52. Song, Q., & Chissom, B. S. (1993). Forecasting enrollments with fuzzy time series — Part I. *Fuzzy Sets and Systems*, *54*(1), 1–9. https://doi.org/10.1016/0165-0114(93)90355-l

53. Song, Q., & Chissom, B. S. (1993). Fuzzy time series and its models. *Fuzzy Sets and Systems*, *54*(3), 269–277. https://doi.org/10.1016/0165-0114(93)90372-o

54. Song, Q., & Chissom, B. S. (1994). Forecasting enrollments with fuzzy time series — part II. *Fuzzy Sets and Systems*, *62*(1), 1–8. https://doi.org/10.1016/0165-0114(94)90067-1

55. Sun, B., Guo, H., Karimi, H. R., Ge, Y., & Xiong, S. (2015). Prediction of stock index futures prices based on fuzzy sets and multivariate fuzzy time series. *Neurocomputing*, *151*(3), 1528–1536. https://doi.org/10.1016/j.neucom.2014.09.018

56. Woo, G., Liu, C., Sahoo, D., Kumar, A., & Hoi, S. (2022). *DeepTIMe: Deep Time-Index Meta-Learning for Non-Stationary Time-Series Forecasting*. arXiv. https://arxiv.org/abs/2207.06046

57. Wu, H., Xu, J., Wang, J., & Long, M. (2021). Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, & J. Wortman Vaughan (Eds.), *Advances in Neural Information Processing Systems: Vol. 34* (pp. 22419-22430). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2021/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf

58. Yahoo. (2022). *Yahoo! Finance* [Data set]. Retrieved September 15, 2022, from https://www.finance.yahoo.com

59. Yolcu, O. C., Yolcu, U., Egrioglu, E., & Aladag, C. H. (2016). High order fuzzy time series forecasting method based on an intersection operation. *Applied Mathematical Modelling*, *40*(19–20), 8750–8765. https://doi.org/10.1016/j.apm.2016.05.012

60. Yu, T. H. (2005). Weighted fuzzy time series models for TAIEX forecasting. *Physica A: Statistical Mechanics and Its Applications*, *349*(3–4), 609–624. https://doi.org/10.1016/j.physa.2004.11.006

61. Yu, T. H.-K., & Huarng, K.-H. (2008). A bivariate fuzzy time series model to forecast the TAIEX. *Expert Systems with Applications*, *34*(4), 2945–2952. https://doi.org/10.1016/j.eswa.2007.05.016

62. Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, *8*(3), 338–353. https://doi.org/10.1016/s0019-9958(65)90241-x

63. Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning—I. *Information Sciences*, *8*(3), 199–249. https://doi.org/10.1016/0020-0255(75)90036-5

64. Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence: Vol. 35(12)* (pp. 11106-11115). AAAI Press. https://doi.org/10.1609/aaai.v35i12.17325

The article was submitted on 2022, September 17