CrossMark

ORIGINAL PAPER

Evolving granular analytics for interval time series forecasting

Leandro Maciel[1] · Rosangela Ballini[2] · Fernando Gomide[1]

Abstract As a paradigm of data processing, granular computation concerns processing complex data entities called granules, which arise from data abstraction and derivation of knowledge from data. This paper addresses granular computation within the framework of interval time series forecasting and evolving intelligent systems. It develops a generalized interval evolving possibilistic fuzzy modeling algorithm as an analytics tool capable to process interval data stream and to produce interval forecasts. The algorithm uses interval arithmetic in its processing steps, employs the notion of data density to adapt the current forecasting model as data are input, and computes (dis)similarity between interval data using the Hausdorff distance. Computational experiments include forecasting of an interval time series data produced by a synthetic time-varying model with parameter drift, and forecasting of financial interval time series using actual daily minimum and maximum values of the US and Brazilian main equity indexes, S&P 500 and IBOVESPA, respectively. The results suggest that the generalized interval evolving possibilistic fuzzy algorithm is highly effective to model and forecast interval time series. It has comparable or better performance than alternative evolving fuzzy and benchmark interval-based approaches.

✉ Leandro Maciel
maciel@dca.fee.unicamp.br

Rosangela Ballini
ballini@eco.unicamp.br

Fernando Gomide
gomide@dca.fee.unicamp.br

[1] School of Electrical and Computer Engineering, University of Campinas, Campinas, São Paulo, Brazil

[2] Institute of Economics, University of Campinas, Campinas, São Paulo, Brazil

1 Introduction

Granular computation (GrC) is a broad-spectrum computational framework that allows different representations of the same problem at different, yet consistent levels of theoretical, methodological and technical details. It serves, *inter alia*, as an infrastructure for knowledge discovery, uncertainty management, data mining, data analytics, decision making, and machine learning.

Granular computation introduces a paradigm to join machine-centric mechanisms for knowledge and data processing with human-centric approaches (Pedrycz 2007). It offers a means to build computational models and algorithms to handle large amount of complex data. Granular computational models aim to process data such as classes, clusters, subsets and intervals.

From the formal point of view, granular computing is structured as a neighborhood system and the notion of nearness (Chen 2009; Lin 2009). The triple structured thinking, structured problem solving, and structured information processing is the view of granular computing from the philosophical, methodological and computational perspective (Yao 2007). Semantical transformation of data and information abstraction are the key issues in the process of granulation and information verification in granular computing models (Bargiela and Pedrycz 2006).

In many granular data processing applications, a piece of data may be viewed as a collection of objects perceived

as a single entity called granule. The objects of a granule are elements of an universe put together by indistinguishability, similarity and functionality (Zadeh 1997). Classes, clusters, subsets and intervals are common examples of granules that appear in computational intelligence and soft computing domains. In data processing and system modeling, granulation of data is often achieved through partition and covering, such as in clustering.

Highly adaptive and intelligent systems, denominated evolving intelligent systems (EIS) (Angelov 2010), is a major subject in contemporary adaptive system modeling and control. In EIS modeling, the model structure and its parameters self-develop simultaneously from a stream of input data coming from time-varying, dynamically changing environments. EIS embody recursive learning and one-pass incremental algorithms that gradually change the model to guarantee life-long learning, and self-organization of the model structure and its parameters (Angelov and Zhou 2008). Self-organization, evolution, and adaptation are important features of complex systems that are largely missing in current studies of granular computation (Chen 2009).

Application of classic statistical methods in time series forecasting often require certain assumptions on the underlying data (Hamilton 1994). Usually, these assumptions are either unrealistic, or are approximations of the true features of the data. In the world of massive, large amount of data and data streams, precision may be expensive, eventually unnecessary. Granular regression appears to be a useful approach in so far traditional statistical regression. Recent approaches include a heuristic method for overlapping granular box regression (Peters 2011), and a generalization of the least squares for granular, non-overlapping boxes (Grzegorzewski 2013). In this context, boxes are vectors of a multidimensional space. An example of box is a multidimensional interval vector formed by the Cartesian product of closed intervals (Jaulin 2001). The role of interval computation as an important part of granular computation is explained in Kreinovich (2008). A comprehensive coverage on how to compute statistics under interval and fuzzy uncertainty is given in Kreinovich (2012).

Prediction of asset prices in financial markets is the basis for asset allocation, portfolio management, risk assessment, and derivatives pricing (Pettenuzzo et al. 2014). The temporal progress of assets, stock indices, and exchange rates is observed as single-valued financial time series (Arroyo et al. 2011). This is useful in many cases, but it may insufficient in situations where several values are observed at each time period. For instance, if only the opening (or closing) asset price is measured daily, the resulting time series will hide the intraday variability and important information is missed. This limitation can be alleviated if the highest and the lowest values of prices are measured at each time period, what originates interval time series (ITS)

(Engleand Russel 2009). Variables of similar nature include electricity prices (Weron 2014), power load and generation, production rates, traffic flows, etc.

In particular, financial ITS modeling and forecasting have received considerable attention in the recent literature.[1] Analysis of large datasets, such as high frequency data, based on the ITS is a new domain to study statistically detectable patterns. It has attracted a large number of researchers in economics and finance (Rodrigues and Salish 2014). In general, the literature addresses ITS using extensions of classic data analytics and forecasting techniques (Fiess and MacDonald 2002; Lima and Carvalho 2008). For instance, Rodrigues and Salish (2014) suggest a univariate threshold model to predict S&P 500 index interval time series independently forecasting the midpoints and half-lengths of the intervals. The authors show that the approach performs better than single-valued time series approaches. Similarly, Arroyo et al. (2011) uses ITS to forecast Dow Jones Index and Euro-Dollar exchange rate using exponential smoothing and artificial neural networks. Forecasts are produced using independent predictions of the lower and upper bounds of intervals or, alternatively, their midpoints and half-lengths. The ITS approach is shown to outperform single-valued methods, specially in exchange rate forecasting.

The vector error correction model (VEC) is a multivariate method Fiess and MacDonald (2002) to forecast interval exchange rate using the lower and the upper interval bounds. Using intraday data, Yang et al. (2014) compares ARIMA, naive, and interval linear regression models to forecast S&P 500, Dow Jones and Nasdaq indexes. The authors claim that the interval-based approaches are superior when compared against single-valued time series models. Recently, Xiong et al. (2014) developed a support vector machine model to simultaneously forecast the minimum and maximum values of the S&P 500, FTSE 100, and Nikkei 225 interval indexes. The results, when compared with classic econometric techniques and interval neural networks, show the superior potential of the interval-based model, especially in financial trading.

The finance and economics related literature has reported the use of evolving fuzzy rule-based models in various areas such as value-at-risk (VaR) modeling and forecasting (Ballini et al. 2009), sovereign bond modeling (Ballini et al. 2009), exchange rate forecasting (McDonald and McDonald 2010), fixed-income option pricing (Maciel 2011), term structure of interest rates (Maciel et al. 2012)

---

[1] The literature has introduced several interval time series forecasting methods. Examples include Lu et al. (2015), Froelichand Salmeron (2014), Xiong et al. (2015) and Wang et al. (2013). The focus of this paper is on financial interval time series forecasting.

and financial volatility forecasting (Luna and Ballini 2012), and stochastic volatility prediction (Luna and Ballini 2012). A recent development is the evolving possibilistic fuzzy modeling approach of Maciel et al. (2015) which gives potentially robust models against noise data and outliers exploring possibilistic clustering (Pal et al. 2005). The evolving possibilisitc approach also employs participatory learning to adapt the model structure, and an utility measure to evaluate its quality.

Despite the recent advances, current interval forecasting approaches have some limitations. For instance: (1) they do not process interval data as a granule. Interval forecasts are produced from the independent forecasts of the intervals bounds or midpoints and half-lengths, which often result in lower performance (He and Hu 2008; 2) multivariate models rely mostly on linear structures, which may not reflect the nonlinear and complex nonstationary dynamics of data properly such as in financial markets for example; (3) they need high amount of data to estimate the parameters of the forecasting models, and are not suitable for stream data processing; (4) data distribution must satisfy certain statistical assumptions; and (5) the imprecise nature of the data are neglected.

This paper addresses interval time series within the framework of granular computation and evolving intelligent systems, focusing on financial interval time series forecasting applications. The aim is to lessen the limitations that current ITS forecasting approaches have. It suggests the generalized interval evolving possibilistic fuzzy modeling (gPFM) as an analytics tool to process interval input stream data and to produce corresponding interval forecasts. More precisely, the paper develops a generalized evolving Takagi–Sugeno (gTS) modeling approach whose rule base and the parameters of the rule consequents are updated as interval data are input. An evolving possibilisitc fuzzy interval data clustering algorithm and weighted recursive least squares (wRLS) compose the two main components of the evolving possibilistic fuzzy forecasting framework. Evolving gTS modeling develops local granular models in fuzzy regions encapsulated by fuzzy clusters with rule consequents as functions of the interval inputs. Model complexity is captured through the combination of local affine interval models of the gTS multi-model structure. Differently from the current financial ITS modeling approaches, approach uses interval arithmetic (Moore et al. 2009; Hickey et al. 2001). As in Maciel et al. (2015), gTS processes interval-valued stream data, but uses an adaptation mechanism based on the notion of potential instead of participatory learning, and the Hausdorff distance to compute the (dis)similarity between intervals instead of Euclidean distance. Computational results are reported using a synthetic interval time series data produced by a time-varying model with parameter drift. Next, financial interval time series forecasting using actual daily minimum and maximum values of the US and Brazilian main equity indexes, S&P 500 and IBOVESPA, respectively, are addressed. The interval time series are the daily minimum and maximum values of the indexes. Performance of the gPFM against ARMA, VEC, evolving fuzzy, and neuro-fuzzy models is evaluated considering the forecasting error of the interval bounds individually, and intervals dissimilarity measure.

The paper proceeds as follows. Section 2 gives a brief reminder of the interval arithmetic adopted in this work. Section 3 details the generalized interval evolving possibilistic fuzzy modeling algorithm. Computational results and performance evaluation are summarized in Sect. 4. Section 5 concludes the paper and suggests issues for further development.

2 Interval arithmetic

In this paper an interval x is a closed bounded set of real numbers:

$$x = [x^L, x^U] \in \mathfrak{I}, \tag{1}$$

where $\mathfrak{I} = \{[x^L, x^U] \; x^L, x^U \in \mathfrak{R}, x^L \leq x^U\}$ is the set of closed intervals of the real line $\mathfrak{R}$, x^L the lower bound, and x^U the upper bound of the interval. An m-dimensional interval vector $\mathbf{x}$ is an ordered m-tuple of intervals $\mathbf{x} = [x_1, x_2, \ldots, x_m]^T$, where $x_j = [x_j^L, x_j^U] \in \mathfrak{I}, j = 1, \ldots, m$.

The midpoint of an interval x is:

$$mp(x) = \frac{x^L + x^U}{2}. \tag{2}$$

An interval time series (ITS), $\{\mathbf{x}_t\}$, is a sequence of interval vectors observed in successive time steps $t = 1, 2, \ldots, T$.

Interval arithmetic extends traditional arithmetic to operate on intervals. This paper uses the arithmetic operations introduced by Moore et al. (2009):

$$\begin{aligned}
x + y &= [x^L + y^L, x^U + y^U], \\
x - y &= [x^L - y^U, x^U - y^L], \\
xy &= [\min\{x^L y^L, x^L y^U, x^U y^L, x^U y^U\}, \\
&\qquad \max\{x^L y^L, x^L y^U, x^U y^L, x^U y^U\}], \\
x/y &= x(1/y), \quad \text{with } 1/y = [1/y^U, 1/y^L].
\end{aligned} \tag{3}$$

Interval arithmetic subsumes classic arithmetic. This means that if an operation of interval arithmetic takes real numbers as operands, considering them as intervals of length zero, then we obtain the same result as if the operation were performed using traditional arithmetic.

The interval evolving possibilistic fuzzy model requires a metric to measure distances and (dis)similarities between

intervals. This paper uses the Hausdorff distance instead of the Euclidean distance (Maciel et al. 2015). If $\mathbf{x}$ and $\mathbf{y}$ are vectors of intervals, then the Hausdorff distance between $\mathbf{x}$ and $\mathbf{y}$, denoted by $d(\mathbf{x}, \mathbf{y})$, is

$$d(\mathbf{x}, \mathbf{y}) = \sum_{j=1}^{m} \left(\max \left\{ |x_j^{\mathrm{L}} - y_j^{\mathrm{L}}|, |x_j^{\mathrm{U}} - y_j^{\mathrm{U}}| \right\} \right). \tag{4}$$

The next section addresses gPFM, the generalized interval evolving possibilistic fuzzy modeling approach. It details its two main components, the generalized evolving Takagi–Sugeno (gTS), and the weighted recursive least squares (wRLS).

3 Generalized interval evolving possibilistic fuzzy modeling

3.1 Generalized Takagi–Sugeno fuzzy model

The generalized Takagi–Sugeno (gTS) fuzzy model with affine interval consequents consists of a set of fuzzy functional rules of the following form:

$$\mathcal{R}_i : \text{IF } \mathbf{x} \text{ is } \mathcal{A}_i \text{ THEN } y_i = \theta_{i0} + \theta_{i1}x_1 + \cdots + \theta_{im}x_m, \tag{5}$$

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, $\mathbf{x} = [x_1, x_2, \ldots, x_m]^T$, $x_j \in \mathfrak{I}$, $j = 1, \ldots, m$, is the input, $\mathcal{A}_i$ is the fuzzy set of the antecedent of the i-th fuzzy rule whose membership function is $\mathcal{A}_i(\mathbf{x}) : \mathfrak{I} \to [,]$, $y_i \in \mathfrak{I}$ is the output of the i-th rule, and θ_{i0} and $\theta_{ij} \in \mathfrak{R}$, $j = 1, \ldots, m$, are real-valued parameters of the consequent of the i-th rule.

Fuzzy inference using gTS rules (5) is similar to the classic Takagi–Sugeno model counterpart except that the arithmetic operations are the interval operations of Sect. 2 instead of the usual operations with real numbers. It is as follows:

$$y = \sum_{i=1}^{c} \left(\frac{\mathcal{A}_i(\mathbf{x}) y_i}{\sum_{j=1}^{c} \mathcal{A}_j(\mathbf{x})} \right). \tag{6}$$

The expression (6) can be rewritten using normalized degrees of activation:

$$y = \sum_{i=1}^{c} \lambda_i y_i = \sum_{i=1}^{c} \lambda_i \mathbf{x}_e^T \theta_i, \tag{7}$$

where

$$\lambda_i = \frac{\mathcal{A}_i(\mathbf{x})}{\sum_{j=1}^{c} \mathcal{A}_j(\mathbf{x})}, \tag{8}$$

is the normalized firing level of the i-th rule, $\theta_i = [\theta_{i0}, \theta_{i1}, \ldots, \theta_{im}]^T$ is the vector of parameters, and $\mathbf{x}_e = [1 \ \mathbf{x}^T]^T$ is the expanded input vector.

gTS modeling requires: (1) learning the antecedent part of the model via an interval fuzzy clustering algorithm, and (2) estimation of the parameters of the interval affine consequents. The i-th fuzzy cluster defines $\mathcal{A}_i$, the antecedent of the i-th fuzzy rule. The cluster structure defines the structure of the model itself because to each cluster corresponds a fuzzy rule whose consequent is, as (5) shows, an interval affine function, an affine local model.

3.2 Interval possibilistic fuzzy clustering

The interval possibilistic fuzzy clustering algorithm is a translation of the possibilistic fuzzy c-means (Pal et al. 2005) to process interval data. Let $\mathbf{x}_k = [x_{1k}, x_{2k}, \ldots, x_{mk}]^T$ be the input data at step k. A set of n inputs is denoted by $X = \{\mathbf{x}_k | k = 1, 2, \ldots, r, \ldots, n\}$. The aim of clustering is to partition the dataset X into c subsets called clusters.

A possibilistic fuzzy partition of the set X is a family $\{\mathcal{A}_i | 1 \le i \le c\}$. Each $\mathcal{A}_i$ characterized by membership degrees and typicalities specified by the fuzzy and typicality partition matrices $U = [u_{ik}] \in [0, 1]^{c \times n}$ and $T = [t_{ik}] \in [0, 1]^{c \times n}$, respectively. The entries of the i-th row of matrix U (T) are the values of membership (typicalities) degrees of the data X in $\mathcal{A}_i$.

The interval possibilistic fuzzy (iPFCM) clustering algorithm produces c vectors of intervals that represent c clusters centers in the interval data space. The iPFCM algorithm may be formulated as the following optimization problem:

$$\min_{U,T,V} \left\{ J = \sum_{k=1}^{n} \sum_{i=1}^{c} (a u_{ik}^{\eta_f} + b t_{ik}^{\eta_p}) d_{ik}(\mathbf{x}_k, \mathbf{v}_i) + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1 - t_{ik})^{\eta_p} \right\}, \tag{9}$$

subject to

$$\sum_{i=1}^{c} u_{ik} = 1 \ \forall \ k, \tag{10}$$
$$0 \le u_{ik}, t_{ik} \le 1.$$

Here $a > 0$, $b > 0$, and $\eta_f, \eta_p > 1$, $\gamma_i > 0$ are parameters chosen by the user, and $d(\cdot, \cdot)$ is a distance measure which, similarly as in Carvalho et al. (2006), is the Hausdorff distance given in (4). The constants a and b define the relative importance of membership degrees and typicality values in the objective function, respectively. $V = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c]^T$ is the matrix of cluster centers, $\mathbf{v}_i = [v_{i1}, v_{i2}, \ldots, v_{im}]^T$, $v_j \in \mathfrak{I}$, $j = 1, \ldots, m$, η_f and η_p are parameters associated with membership degrees and typicalities (usually $\eta_f = \eta_p = 2$).

In this paper, membership degrees and typicalities are found using:

$$u_{ik} = \left(\sum_{j=1}^{c} \left(\frac{d_{ik}(\mathbf{x}_k, \mathbf{v}_i)}{d_{jk}(\mathbf{x}_k, \mathbf{v}_j)} \right)^{2/(\eta_f - 1)} \right)^{-1}, \tag{11}$$

$$t_{ik} = \frac{1}{1 + \left(\frac{b}{\gamma_i} d_{ik}^2(\mathbf{x}_k, \mathbf{v}_i) \right)^{1/(\eta_p - 1)}}, \tag{12}$$

and the cluster centers approximated by:

$$\mathbf{v}_i = \frac{\sum_{k=1}^{n} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p})\mathbf{x}_k}{\sum_{k=1}^{n} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p})}. \tag{13}$$

Strictly speaking, cluster centers derived as a solution of (9) requires computation of medians of intervals (Carvalho and Pimentel 2012) which in turn, since the median is nondecreasing, can be computed as the medians of the lower and upper bounds of intervals. Despite this fact, because a recursive expression to compute the median is difficult to obtain, and we rely on (13) as a heuristic approximation inspired in Kratschmer (2002).

A way to choose γ_i is as follows (Pal et al. 2005):

$$\gamma_i = K \frac{\sum_{k=1}^{n} u_{ik}^{\eta_f} d_{ik}(\mathbf{x}_k, \mathbf{v}_i)}{\sum_{k=1}^{n} u_{ik}^{\eta_f}}, \quad 1 \le i \le c, \tag{14}$$

where $K > 0$ (usually $K = 1$), and u_{ik} are entries of a terminal FCM partition of X.

3.3 Recursive interval possibilistic fuzzy clustering

Generalized interval evolving possibilistic fuzzy modeling relies on the recursive form of the possibilistic fuzzy clustering for interval data (Maciel et al. 2015).[2]

Let $\mathbf{v}_{ir} = [v_{1,ir}, v_{2,ir}, \ldots, v_{m,ir}]^T$ be the i-th cluster center at step r, using the values of membership degrees and typicality given by (11) and (11). The i-th cluster center at step $r + 1$ becomes:

$$\mathbf{v}_{i,r+1} = \frac{\sum_{k=1}^{r+1} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p})\mathbf{x}_k}{\sum_{k=1}^{r+1} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p})}$$

$$= \frac{\sum_{k=1}^{r} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p})\mathbf{x}_k + (au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})\mathbf{x}_{r+1}}{\sum_{k=1}^{r} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p}) + (au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})}. \tag{15}$$

Denote the relation between the cluster center at step r and the new one at step $r + 1$ by:

$$\mathbf{v}_{i,r+1} = \mathbf{v}_{ir} + \Delta\mathbf{v}_{i,r+1}, \tag{16}$$

where $\Delta\mathbf{v}_{i,r+1}$ is the variation in the cluster center due to new input data $\mathbf{x}_{r+1}$. According to Maciel et al. (2015), using (15), expression (16) can be rewritten as follows:

$$\mathbf{v}_{i,r+1} = \mathbf{v}_{ir} + \frac{(au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})(\mathbf{x}_{r+1} - \mathbf{v}_{ir})}{\sum_{k=1}^{r} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p}) + (au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})}. \tag{17}$$

The increment of the i-th cluster center $\Delta\mathbf{v}_{i,r+1}$ is:[3]

$$\Delta\mathbf{v}_{i,r+1} = \frac{(au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})(\mathbf{x}_{r+1} - \mathbf{v}_{ir})}{\sum_{k=1}^{r} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p}) + (au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})}. \tag{18}$$

The denominator of cluster centers increments (18) needs the values of all past r data. In principle, the denominator can also be computed recursively. However, in adaptive modeling and non-stationary environments past data may become obsolete and current data should be emphasized because they carry more accurate information about the current state. A suitable mechanism to tackle this situation is to use exponential smoothing to weight past membership and typicality degrees.

Let the sum of the memberships and typicalities up to step r in the denominator of (18) be $s_{ik} \in \Re$:

$$s_{ir} = \sum_{k=1}^{r} (au_{ik}^{\eta_f} + bt_{ik}^{\eta_p}). \tag{19}$$

After new data is input at step $r + 1$, the value of $s_{i,r+1}$ can be estimated using exponential smoothing as follows:

$$s_{i,r+1} = \gamma_v s_{ir} + au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p}. \tag{20}$$

The parameter γ_v, $0 \le \gamma_v \le 1$ is the forgetting factor that weights past membership degrees. Thus $\Delta\mathbf{v}_{i,r+1}$ can be rewritten as:

$$\Delta\mathbf{v}_{i,r+1} = \frac{(au_{i,r+1}^{\eta_f} + bt_{i,r+1}^{\eta_p})(\mathbf{x}_{r+1} - \mathbf{v}_{ir})}{s_{i,r+1}}. \tag{21}$$

After computation of cluster centers, the values of the membership degrees $u_{i,r+1}$ and typicalities $t_{i,r+1}$ of input data $\mathbf{x}_{r+1}$ at step $r + 1$ are computed using (11) and (12), that is:

$$u_{i,r+1} = \left(\sum_{j=1}^{c} \left(\frac{d_{i,r+1}(\mathbf{x}_{r+1}, \mathbf{v}_i)}{d_{j,r+1}(\mathbf{x}_{r+1}, \mathbf{v}_j)} \right)^{2/(\eta_f - 1)} \right)^{-1}, \tag{22}$$

$$t_{i,r+1} = \frac{1}{1 + \left(\frac{b}{\gamma_i} d_{i,r+1}^2(\mathbf{x}_{r+1}, \mathbf{v}_i) \right)^{1/(\eta_p - 1)}}. \tag{23}$$

---

[2] Chuang et al. (2013) also suggests an interval fuzzy possibilistic c-means clustering algorithm based on Hausdorff distance, but for batch data processing only.

[3] Detailed derivation of the recursive expression of cluster centers is found in Maciel et al. (2015).

3.4 Model structure adaptation

In this paper, the mechanism to adapt the rule-based model uses the concept of recursive data density estimation. Contrary to the participatory learning approach of Maciel et al. (2015), recursive data density does not require choice of threshold parameters. The density of data $\mathbf{x}_k$ at step k, $\mathcal{D}_k(\mathbf{x}_k)$, can be estimated using a Cauchy function:

$$\mathcal{D}_k(\mathbf{x}_k) = \frac{1}{1 + \frac{1}{k-1}\sum_{j=1}^{k-1}\sum_{l=1}^{m}(x_{k,l} - x_{j,l})^2}. \tag{24}$$

Evolving systems need the density evaluated recursively to accumulate the information about spatial distribution of all data. The recursive calculation of data density is as follows (Angelov 2010):

$$\mathcal{D}_k(\mathbf{x}_k) = \frac{k-1}{(k-1)\left(\sum_{l=1}^{m} x_{k,l}^2 + 1\right) + h_k - 2\sum_{l=1}^{m} x_{k,j}p_{k,j}}, \tag{25}$$

where:

$$h_k = h_{k-1} + \sum_{l=1}^{m}(x_{k-1,l})^2, h_1 = 0, \tag{26}$$

$$p_{k,l} = p_{k-1,l} + x_{k-1,l}, p_{1,l} = 0.$$

Data with high density are potential candidates to be cluster prototypes. If a data point is selected to be a cluster center, then it becomes a focal point (25) and has the value of its density stored. The density of cluster prototypes is updated using:

$$\mathcal{D}_k(\mathbf{x}^{i^*}) = \frac{k-1}{k-1 + (k-2)\left(\frac{1}{\mathcal{D}_{k-1}(\mathbf{x}^{i^*})} - 1\right) + \sum_{l=1}^{m}(x_l^{i^*} - x_{k,l})^2}, \tag{27}$$

where $\mathcal{D}_1(\mathbf{x}^{i^*}) = 1$ and i^* denotes the centroid of the i-th cluster.

New clusters may formed by data with high density using the following condition (Angelov 2010):

$$\text{IF } \mathcal{D}_k(\mathbf{x}_k) > \max_{i=1}^{c} \mathcal{D}_k(\mathbf{x}^{i^*}) \text{ THEN } \mathbf{x}^{(c+1)^*} \leftarrow \mathbf{x}_k, \quad c \leftarrow c+1. \tag{28}$$

The quality of the cluster structure is monitored at each step considering the utility measure suggested in Angelov (2010). The utility measure is an indicator of the accumulated relative firing level of a rule:

$$\mathcal{U}_{ik} = \frac{\sum_{l=1}^{k} \lambda_i}{k - I^{i*}}, \tag{29}$$

where I^{i*} is the step (time tag) that indicates when the corresponding fuzzy rule $i*$ was created. Once a cluster, or equivalently, a rule is created, the utility indicates how much the rule has been used so far. The quality measure aims at avoiding unused clusters kept in the structure. Clusters with corresponding low quality fuzzy rules can be deleted. In other words, the following criterion is used to delete a cluster/rule:

$$\text{IF } \mathcal{U}_{ik} < \epsilon \text{ THEN } c \leftarrow c - 1, \tag{30}$$

where $\epsilon \in [0.03; 0.1]$ is a threshold that controls the utility level of each cluster (Angelov 2010). This principle guarantees high relevance cluster structures and local models. Alternative quality measures such as age, support, zone of influence and local density may be adopted.

3.5 Estimation of the model parameters

The parameters of the interval affine consequents of the rule-based model (5) are estimated using weighted recursive least squares algorithm (wRLS) (Ljung 1988). To take advantage of the standard form of the wRLS algorithm, the procedure uses the midpoint $\text{mp}(x_j)$ of the intervals as in (2). Consider expression (7) rewritten as:

$$\text{mp}(y) = \Lambda^T \Theta, \tag{31}$$

where $\Lambda = \left[\lambda_1 \text{mp}(\mathbf{x}_e^T), \lambda_2 \text{mp}(\mathbf{x}_e^T), \ldots, \lambda_c \text{mp}(\mathbf{x}_e^T)\right]^T$ is the fuzzily weighted extended input, $\text{mp}(\mathbf{x}_e) = [1 \ \text{mp}(\mathbf{x}^T)]^T$ is the expanded data vector, and $\Theta = \left[\theta_1^T, \theta_2^T, \ldots, \theta_c^T\right]^T$ is the parameter matrix, $\theta_i = [\theta_{i0}, \theta_{i1}, \ldots, \theta_{im}]^T$.

Given that the actual output is known at each step, the parameters of the consequents can be updated using the weighted recursive least squares (wRLS) algorithm considering either local or global optimization. This paper employs the locally optimal error criterion wRLS:

$$\min E_L^i = \min \sum_{k=1}^{n} \lambda_i \left(\text{mp}(y_k) - \text{mp}(\mathbf{x}_{ek}^T)\theta_{ik}\right)^2. \tag{32}$$

The parameters of the rule consequents are updated as follows (Ljung 1988):

$$\theta_{i,k+1} = \theta_{ik} + \Sigma_{ik}\text{mp}(\mathbf{x}_{ek})\lambda_{ik}\left(\text{mp}(y_k) - \text{mp}(\mathbf{x}_{ek}^T)\theta_{ik}\right), \quad \theta_{i0} = 0, \tag{33}$$

$$\Sigma_{i,k+1} = \Sigma_{ik} - \frac{\lambda_{ik}\Sigma_{ik}\text{mp}(\mathbf{x}_{ek})\text{mp}(\mathbf{x}_{ek}^T)\Sigma_{ik}}{1 + \lambda_{ik}\text{mp}(\mathbf{x}_{ek}^T)\Sigma_{ik}\text{mp}(\mathbf{x}_{ek})}, \quad \Sigma_{i0} = \Omega I, \tag{34}$$

where I is the $(m + 1)$ identity matrix; Ω is a large number (usually $\Omega = 1000$), and Σ is the dispersion matrix.

3.6 gPFM algorithm

The generalized interval evolving possibilistic fuzzy modeling (gPFM) algorithm is summarized in this section. The steps of the algorithm are non-iterative. The procedure

revises an existing model whenever there is new information in the input data. The recursive nature of gPFM turns it particularly useful for on-line and real-time applications.

Assume that an initial dataset is available. The first step of the algorithm is to partition the initial dataset using the FCM clustering algorithm (step 1) to get γ_i and $\mathbf{v}_{i0}$. The next step is to select the control parameters (step 2) and to initialize s_{i0} for the existing c rules (step 3). Next, gPFM starts processing input data stream (step 4) and for each existing rule, it computes the density of the input data (step 6) and updates the density of the focal points (step 7). If the density of the current input data is higher than the maximum density of all existing focal points, then a new rule is created (steps 8–10). Otherwise, the rule base remains the same as in the previous step. Membership degrees and typicality values are computed (steps 12 and 13) using the distance from the current input data to all the existing clusters centers (step 11). Next, the clusters centers are updated accordingly (steps 14–16). The utility of the rule is evaluated and if it is lower than the threshold, then it is deleted from the rule base (steps 17–19). Once the cluster structure is evaluated (steps 5–20), the parameters of rule consequents are updated using wRLS (step 21) and the model output is computed (step 22) using (7).

Generalized intervalar evolving possibilistic fuzzy modeling

1.   get γ_i and $\mathbf{v}_{i0}$ from a terminal FCM partition
2.   choose values for γ_v, a, b, ϵ
3.   set $s_{i0} = \sum_{k=1}^{r} a u_{ik}^{\eta_f} + b t_{ik}^{\eta_p}$
4.   **for** $k = 1, 2, \ldots$ **do**
5.   **for** $i = 1, 2, \ldots c$ **do**
6.   compute $\mathcal{D}_k(\mathbf{x}_k) = \frac{k-1}{(k-1)\left(\sum_{l=1}^{m} x_{k,l}^2 + 1\right) + h_k - 2\sum_{l=1}^{m} x_{k,j}p_{k,j}}$
7.   compute $\mathcal{D}_k(\mathbf{x}^{i^*}) = \frac{k-1}{k-1+(k-2)\left(\frac{1}{\mathcal{D}_{k-1}(\mathbf{x}^{i^*})}-1\right)+\sum_{l=1}^{m}\left(x_l^{i^*}-x_{k,l}\right)}$
8.   **if** $\mathcal{D}_k(\mathbf{x}_k) > \max_{i=1}^{c} \mathcal{D}_k(\mathbf{x}^{i^*})$ **then** create a new cluster
9. $c \leftarrow c + 1$
10.   **end if**
11.   compute $d_{ik}(\mathbf{x}_k, \mathbf{v}_i) = \sum_{j=1}^{m}\left(\max\left\{|x_j^L - y_j^L|, |x_j^U - y_j^U|\right\}\right)$
12.   update $u_{ik} = \left(\sum_{j=1}^{c}\left(\frac{d_{ik}(\mathbf{x}_k,\mathbf{v}_i)}{d_{jk}(\mathbf{x}_k,\mathbf{v}_j)}\right)^{2/(\eta_f-1)}\right)^{-1}$
13.   update $t_{ik} = \frac{1}{1+\left(\frac{b}{\gamma_i}d_{ik}^2(\mathbf{x}_k,\mathbf{v}_i)\right)^{1/(\eta_p-1)}}$
14.   compute $s_{ik} = \gamma_v s_{i,k-1} + a u_{ik}^{\eta_f} + b t_{ik}^{\eta_p}$
15.   compute $\Delta \mathbf{v}_{ik} = \frac{(a u_{ik}^{\eta_f} + b t_{ik}^{\eta_p})(\mathbf{x}_k - \mathbf{v}_{i,k-1})}{s_{ik}}$
16.   update centers $\mathbf{v}_{ik} = \mathbf{v}_{i,k-1} + \Delta \mathbf{v}_{i,k}$
17.   **if** $\mathcal{U}_{ik} < \epsilon$ **then** delete cluster i
18. $c \leftarrow c - 1$
19.   **end if**
20.   **end for**
21.   update rule consequent parameters using wRLS
22.   compute model output y_{k+1}
23.   **end for**

---

4 Computational experiments

The interval gPFM approach introduced in this paper is a flexible modeling procedure useful for a variety of problems. This section illustrates interval time series

forecasting using gPFM to model synthetic data with concept drift and actual financial interval data.

4.1 Data

First, gPFM is run to model a time-varying nonlinear process whose model is an extension of the one suggested in Narendra and Parthasarathy (1990). The process model produces a time series with concept drift (TSCD) using the expression:

$$\tilde{y}_{k+1} = \frac{\delta_1 \tilde{y}_k}{1 + \delta_2 \tilde{y}_k^2} + \delta_3 \tilde{x}_k^3, \tag{35}$$

where $\tilde{x}_k = \sin(2\pi k/25) + \sin(2\pi k/10)$, $\tilde{y}_k = 0$ for $k \le 0$, $\tilde{y} \in \Re$ and $\tilde{x} \in \Re$.

The goal is to develop a model $f(\cdot)$ of the form:

$$\tilde{y}_k = f(\tilde{y}_{k-1}, \tilde{y}_{k-2}, \tilde{y}_{k-3}, \tilde{y}_{k-4}). \tag{36}$$

The values of the parameters δ_1, δ_2 and δ_3 change at $k = 300$ as follows (Narendra and Parthasarathy 1990):

$$\delta_0 = 0.5\sin(0.2k), \tag{37}$$

$$\delta_1 = \delta_0 + \begin{cases} 1 & \text{if } k < 200 \\ 3.5 & \text{otherwise,} \end{cases} \tag{38}$$

$$\delta_2 = \delta_0 + \begin{cases} 1 & \text{if } k < 200 \\ 0.8 & \text{otherwise,} \end{cases} \tag{39}$$

$$\delta_3 = \delta_0 + \begin{cases} 1 & \text{if } k < 200 \\ 1.5 & \text{otherwise.} \end{cases} \tag{40}$$

Data consists of a stream of 440 samples, the first 240 samples for learning, and the remaining 200 for testing.

The interval time series (iTSCD) is formed by subtracting and adding 20 % of TSCD to itself to produce the lower (y_k^L) and upper bounds (y_k^U) at each step k of the corresponding interval time series iTSCD as follows:

$$y_k^L = \tilde{y}_k - (0.2\tilde{y}_k), \tag{41}$$

$$y_k^U = \tilde{y}_k + (0.2\tilde{y}_k). \tag{42}$$

Second, the gPFM is run to process financial interval time series forecasting using the main US and Brazil equity market indexes, S&P 500 and IBOVESPA, respectively. Daily interval time series data are formed using the daily minimum and maximum values of the S&P 500 and IBOVESPA indexes from January 3, 2000 to December 28, 2012.[4] Each daily interval datum has the daily minimum index value as the lower bound, and the daily maximum index value as the upper bound. Time series data were split into two sets, the first for learning, and the second for testing. Two different cut-off data were considered to evaluate the performance of gPFM. Learning dataset (I) refers to the ITS from January 3, 2000 to

---

[4] The data provided by Bloomberg.

Table 1 Performance evaluation for iTSCD forecast

Models	RMSE^L	RMSE^U	EMDE	# rules (aver.)	Time (s)
eTS (Angelov and Filev 2004)	0.374	0.389	6.312	4.19	3.12
xTS (Angelov and Zhou 2008)	0.369	0.374	6.274	3.89	3.04
eTS+ (Angelov 2010)	0.350	0.363	6.195	3.73	2.78
ePL+ (Maciel et al. 2014)	0.358	0.369	6.316	4.08	2.89
eFuMo (Dovžan et al. 2012)	**0.312**	0.319	5.987	**3.64**	**2.39**
DENFIS (Kasabov and Song 2002)	0.362	0.377	6.414	8	41
gPFM	0.315	**0.308**	**4.980**	3.95	2.51

Bold values represent best results

December 31, 2007 whereas training dataset (II) refers to the ITS from January 3, 2000 to December 31, 2008. Test dataset (I) are the remaining data, that is, from January 2, 2008 to December 28, 2012. Similarly, test dataset (II) is from January 2, 2009 to December 28, 2012.

4.2 Performance evaluation

The results of the generalized intervalar evolving possibilistic fuzzy modeling (gPFM) are compared with evolving fuzzy and neuro-fuzzy modeling methodologies, respectively, eTS, xTS, eTS+, ePL+, eFuMo and DENFIS. The results of the financial ITS include comparisons with the ARIMA and VEC models. VEC models are benchmarks for financial ITS approaches because they are multivariate methods that simultaneously estimate the intervals bounds. We notice that all the remaining approaches, except the interval gPFM, forecast intervals bounds independently of each other. More precisely, they forecast the lower and upper values of the intervals bounds viewing them as independent, single-valued time series.

Forecast is one step ahead.[5] Performance of the approaches are evaluated and compared in terms of intervals bounds accuracy and interval distance error. The evaluation considers forecasts of the minimum (lower bound) and maximum (upper bound) values of the ITS using the root mean square error (RMSE$^{L/U}$):

$$\text{RMSE}^{L/U} = \left(\frac{1}{n}\sum_{k=1}^{n}(d_k^{L/U} - y_k^{L/U})^2\right)^{\frac{1}{2}}, \quad (43)$$

where $d_k^{L/U}$ is the actual lower/upper interval bound at k, $y_k^{L/U}$ is the forecast lower/upper interval bound at k, and n is the sample size.

An important performance measure of ITS is to compute the forecasting error as the distance between the actual

interval data and the forecast interval. This paper evaluates forecast performance using the Euclidean mean distance error (EMDE) Arroyo et al. (2011):

$$\text{EMDE} = \left(\frac{1}{n}\sum_{k=1}^{n}\left[(d_k^L - y_k^L)^2 + (d_k^U - y_k^U)^2\right]\right)^{\frac{1}{2}}, \quad (44)$$

where $d_k = [d_k^L, d_k^U]$ and $y_k = [y_k^L, y_k^U]$ are the actual and forecasted interval data, respectively, at k.

The complexity of the models are evaluated in terms of the average number of rules/nodes, and the (CPU) time elapsed during test data processing. All algorithms were implemented and run using Matlab® on a laptop with 4 GB and Intel ®i3CPU.

4.3 Forecasting interval time series with concept drift iTSCD

Concerning the interval time series with concept drift, simulations conducted to find the best performance suggested the following control parameters for gPFM: $\gamma_v = 0.93$, $a = 1$, $b = 3$, and $\epsilon = 0.10$. Initialization uses the FCM algorithm to choose γ_i and $\mathbf{v}_{i0}$. Similarly, the best control parameters of the alternative models were also chosen by simulations.

Table 1 shows the RMSE and EMDE values of interval iTSCD modeling achieved by gPFM and alternative approaches using test data. The lowest values are
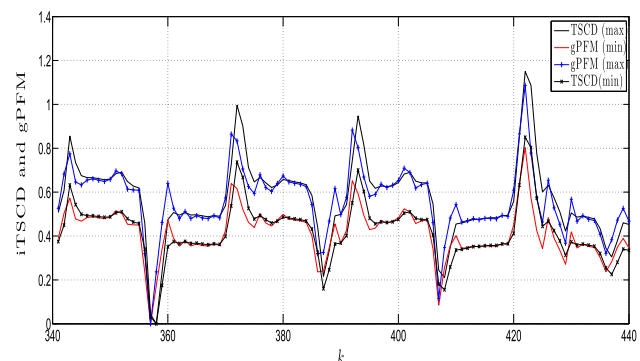
---

[5] gPFM is also able to perform forecasts for h steps ahead, i.e., to produce forecasts for y_{t+h}. The forecasting horizon depends on the application and the user goals. In risk management, one step ahead forecasting are more appealing because of the dynamics of risk assessment.

Fig. 1 Actual iTSCD data and the gPFM model output

Table 2 Performance evaluation for interval S&P 500 index test dataset (I), from January 2, 2008 to December 28, 2012

Models	RMSE^L	RMSE^U	EMDE	# rules (aver.)	Time (s)
ARIMA	0.596	0.610	9.563	–	14
VEC	0.520	0.531	8.209	–	9
eTS (Angelov and Filev 2004)	0.432	0.437	6.892	5.48	2.43
xTS (Angelov and Zhou 2008)	0.427	0.440	6.917	6.11	2.28
eTS+ (Angelov 2010)	0.415	0.418	6.432	5.70	2.87
ePL+ (Maciel et al. 2014)	0.444	0.427	6.498	6.54	**2.26**
eFuMo (Dovžan et al. 2012)	**0.391**	0.407	6.132	**5.09**	2.39
DENFIS (Kasabov and Song 2002)	0.420	0.430	6.564	7	53
gPFM	0.399	**0.398**	**5.877**	5.37	2.44

Bold values represent best results

highlighted. In terms of RMSE considering the lower (RMSE^L) and upper (RMSE^U) interval bounds, the results of gPFM are similar or better when compared against the eFuMo, eTS+ and the remaining models. From the point of view of the distance between intervals EMDE, gPFM gives significant better results than all the approaches (Table 1). eFuMo develops the smallest average number of rules amongst all approaches (Table 1). Small number of rules means smaller model complexity and better interpretability. The models performed similarly in terms of processing time, all of them qualifying to process granular stream data, except DENFIS which requires much higher processing time.

Figure 1 shows the behavior of gPFM modeling of the interval time series with concept drift (iTSCD). The figure depicts the actual data for the last 100 intervals from test data and the output values of gPFM. Notice that the gPFM modeling faithfully captures the nonlinear and time-varying nature of the data.

4.4 Financial interval time series forecasting

Forecasting of S&P 500 and IBOVESPA indexes uses control parameters chosen from simulations conducted to find gPFM best performance, namely, $\gamma_v = 0.90$, $a = 1$, $b = 2$, and $\epsilon = 0.09$ for S&P 500, and $\gamma_v = 0.94$, $a = 2$, $b = 3$, and $\epsilon = 0.10$ for IBOVESPA. The parameters are the same for both cut-off date sets. Initialization uses the FCM algorithm to choose γ_i and $\mathbf{v}_{i0}$. Control parameters of the alternative evolving fuzzy models were also selected from simulation experiments.

Tables 2 and 3 show RMSE and EMDE values of ePFM and alternative algorithms for S&P 500 indexes using test dataset (I), i.e., from January 2, 2008 to December 28, 2012, and using test dataset (II), i.e., from January 2, 2009 to December 28, 2012, respectively. Best results are in bold. Results are quite similar for both cut-off date sets, since test dataset (II) essentially excludes the year of 2008 in comparison to test dataset (I). The high volatility period of 2008 provoked by the recent financial crisis, implies in a

slight higher level of forecasting errors for the test dataset (I) (Table 2) when compared with the test dataset (II) (Table 3).

The RMSE of lower and upper bounds of S&P 500 index forecasted by gPFM show similar magnitudes when compared with eTS+, eFuMo and remaining models (Tables 2, 3). ARIMA and VEC achieve the lowest accuracy, but VEC performs better than ARIMA. From the point of view of EMDE, gPFM gives significant better results than all remaining methodologies. eFuMo develops the smallest average number of rules amongst all competing models. The processing time off all approaches are similar, except ARIMA, VEC and DENFIS, which require higher time to process the test data.

Figure 2 illustrates the performance of gPFM for intervalar S&P 500 index forecasting. It depicts the actual data and the output values by gPFM considering the last year of data from test set (I).[6] It is worth to note that the gPFM modeling approach captures the dynamic of the financial index, even in periods of higher volatility.

Concerning the IBOVESPA index, Tables 4 and 5 show RMSE and EMDE values of ePFM and alternatives modeling of IBOVESPA index using test dataset (I), i.e., from January 2, 2008 to December 28, 2012, and using test dataset (II), i.e., from January 2, 2009 to December 28, 2012, respectively. As for S&P 500, results for both test data sets (I) and (II) are very similar, in which the levels of errors for test dataset (II) (Table 5) are slight small than fro test dataset (I) (Table 4). The gPFM model achieves competitive results in terms of RMSE when compared with eFuMo, DENFIS and the remaining models (Tables 4, 5). The ARIMA and VEC models have the lowest accuracy. The computational performance of the approaches are similar for both, lower (RMSE^L) and upper (RMSE^U) interval bounds, i.e., minimum and

---

[6] Due to the gPFM structure by processing streaming data, the results for the last year of data are very similar for both test data sets (I) and (II). Thus, for the better visual interpretation we kept only the results from test dataset (I) for both S&P 500 and IBOVESPA.

Table 3 Performance evaluation for interval S&P 500 index test dataset (II), from January 2, 2009 to December 28, 2012

Models	RMSE^L	RMSE^U	EMDE	# rules (aver.)	Time (s)
ARIMA	0.605	0.598	9.328	–	13
VEC	0.534	0.516	8.009	–	8
eTS (Angelov and Filev 2004)	0.429	0.428	6.754	5.10	**2.01**
xTS (Angelov and Zhou 2008)	0.419	0.422	6.832	5.78	2.08
eTS+ (Angelov 2010)	0.420	0.408	6.264	5.45	2.45
ePL+ (Maciel et al. 2014)	0.430	0.414	6.387	6.32	2.02
eFuMo (Dovžan et al. 2012)	0.387	0.389	5.980	**4.76**	2.07
DENFIS (Kasabov and Song 2002)	0.398	0.422	6.336	7	49
gPFM	**0.386**	**0.374**	**5.654**	5.19	2.14

Bold values represent best results

Table 4 Performance evaluation for interval IBOVESPA index test dataset (I), from January 2, 2008 to December 28, 2012

Models	RMSE^L	RMSE^U	EMDE	# rules (aver.)	Time (s)
ARIMA	0.690	0.709	8.970	–	19
VEC	0.607	0.631	7.316	–	13
eTS (Angelov and Filev 2004)	0.557	0.577	7.015	6.23	2.60
xTS (Angelov and Zhou 2008)	0.542	0.568	7.129	6.40	2.67
eTS+ (Angelov 2010)	0.539	0.553	6.651	**5.89**	2.71
ePL+ (Maciel et al. 2014)	0.558	0.560	6.762	7.17	2.69
eFuMo (Dovžan et al. 2012)	0.491	**0.497**	6.401	6.19	**2.44**
DENFIS (Kasabov and Song 2002)	0.511	0.521	6.445	10	70
gPFM	**0.459**	0.503	**5.476**	7.44	2.61

Bold values represent best results

Table 5 Performance evaluation for interval IBOVESPA index test dataset (II), from January 2, 2009 to December 28, 2012

Models	RMSE^L	RMSE^U	EMDE	# rules (aver.)	Time (s)
ARIMA	0.612	0.684	8.648	-	18
VEC	0.576	0.601	7.122	-	11
eTS (Angelov and Filev 2004)	0.509	0.547	6.873	6.09	2.32
xTS (Angelov and Zhou 2008)	0.529	0.539	6.932	6.25	2.56
eTS+ (Angelov 2010)	0.513	0.524	6.352	**5.74**	2.54
ePL+ (Maciel et al. 2014)	0.509	0.530	6.498	6.80	2.49
eFuMo (Dovžan et al. 2012)	0.456	**0.476**	6.296	5.78	**2.21**
DENFIS (Kasabov and Song 2002)	0.498	0.498	6.207	10	69
gPFM	**0.438**	0.480	**5.113**	7.20	2.33

Bold values represent best results

maximum values of IBOVESPA index, respectively. eTS+ develops the smallest average number of rules amongst all approaches. Generally speaking, all approaches are qualified to deal with financial interval bounds forecasting. Tables 4 and 5 show IBOVESPA interval time series EMDE values of gPFM and remaining approaches. Clearly, gPFM achieves significant better results than all remaining approaches. The classic ARIMA and VEC econometric benchmarks perform worst, but VEC model is more accurate than ARIMA.

Figure 3 shows the actual data and the forecast output values of gPFM for IBOVESPA index for test dataset (I). Again, one notice the higher performance of gPFM to forecast thge IBOVESPA index.
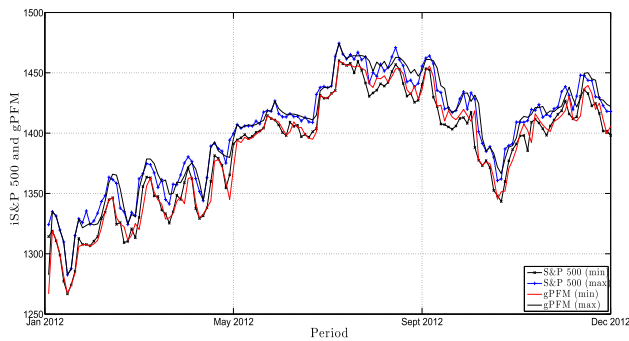
Fig. 2 Actual interval S&P 500 index data and the gPFM forecast for test dataset (I)

Fig. 3 Actual intervalar IBOVESPA index data and the gPFM forecast for test dataset (I)

5 Conclusion

This paper has suggested a generalized evolving possibilisitc fuzzy modeling algorithm as an analytic tool for interval time series forecasting. The approach extends the evolving possibilistic fuzzy modeling methods to processes interval data granules. The performance of gPFM was evaluated using a synthetic interval time series with concept drift, and using actual data of the S&P 500 and IBOVESPA, the main US and Brazilian equity market indexes, respectively. The results show that the gPFM produces better forecasts than the individuals bounds forecasts of ARIMA, VEC, and state of the art evolving fuzzy and neuro-fuzzy approaches. Future work shall evaluate how gPFM perform in high dimensional spaces, and to compare gPFM with existing granular box regression models.

References

Angelov P (2010) Evolving Takagi–Sugeno fuzzy systems from data streams (eTS+). In: Angelov P, Filev D, Kasabov N (eds) Evolving intelligent systems: methodology and applications. Wiley & IEEE Press, Hoboken, pp 21–50

Angelov P, Filev D (2004) An approach to online identification of Takagi–Sugeno fuzzy models. IEEE Trans Syst Man Cybern Part B 34(1):484–498

Angelov P, Zhou X (2008) Evolving fuzzy-rule-based classifiers from data streams. IEEE Trans Fuzzy Syst 16(6):1462–1475

Arroyo J, Espínola R, Maté C (2011) Different approaches to forecast interval time series: a comparison in finance. Comput Econ 27(2):169–191

Ballini R, Mendonça ARR, Gomide F (2009) Evolving fuzzy modeling in risk analysis. Intell Syst Acc Fin Manag 16:71–86

Ballini R, Mendonça ARR, Gomide F (2009) Evolving fuzzy modeling of sovereign bonds. J Fin Decis Mak 16:3–15

Bargiela A, Pedrycz W (2006) The roots of granular computing. In: Proceedings of the 2006 IEEE international conference on granular computing, USA, pp 806–809

Carvalho F, Pimentel J (2012) Partitioning fuzzy clustering algorithms for interval-valued data based on hausdorff distances. In: Proceedings of the 2012 IEEE international conference on systems, man, and cybernetics, Seoul, Korea, pp 14–17

Carvalho FAT, Souza R, Chavent M, Lechevallier I (2006) Adaptive Hausdorff distances and dynamic clustering of symbolic interval data. Pattern Recognit Lett 27(3):167–179

Chen Z (2009) Philosophical foundation for granular computing. Encyclopedia of complexity and systems science. Springer, New York

Chuang C, Jeng J, Chang S (2013) Hausdorff distance measure based interval fuzzy possibilistic c-means clustering algorithm. Int J Fuzzy Syst 15(4):471–479

Dovžan D, Loga V, Škrjanc I (2012) Solving the sales prediction with fuzzy evolving models. In: WCCI 2012 IEEE World congress on computational intelligence, pp 10–15, June, Brisbane, Australia

Engle RF, Russel J (2009) Analysis of high frequency data. In: Aït-Sahalia Y, Hansen LP (eds) Handbook of financial econometrics: tools and techniques, vol 1. North Holland, Amsterdam, pp 383–346

Fiess NM, MacDonald R (2002) Towards the fundamentls of technical analysis: analysing the information content of high, low and close prices. Econ Model 19(3):353–374

Froelich W, Salmeron JL (2014) Evolutionary learning of fuzzy grey cognitive maps for the forecasting of multivariate, interval-valued time series. Int J Approx Reason 55(6):1319–1335

Grzegorzewski P (2013) Granular regression. In: Proceedings of the IFSA/NAFIPS World congress, Edmonton, Canada, pp 974–979

Hamilton J (1994) Time series analysis. Princeton University Press, Princeton

He LT, Hu C (2008) Impacts of interval measurement on studies of economic variability: evidence from stock market variability forecasting. J Risk Fin 8(5):489–507

Hickey T, Ju Q, Emden M (2001) Interval arithmetic: from principles to implementation. J ACM 48(5):1038–1068

Jaulin L (2001) Applied interval analysis. Springer, London

Kasabov NK, Song Q (2002) DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. IEEE Trans Fuzzy Syst 10(2):144–154

Kratschmer V (2002) Limit theorems for fuzzy-random variables. Fuzzy Sets Syst 126:253–263

Kreinovich V (2008) Interval computation as an important part of granular computing: an introduction. In: Pedrycz W, Skowron W, Kreinovich V (eds) Handbook of granular computing. Wiley, West Sussex

Kreinovich V (2012) Computing statistics under interval and fuzzy uncertainty: applications to computer sciene and engineering. Springer, Heidelberg

Lima EA, Carvalho FATD (2008) Centre and range method for fitting a linear regression model to symbolic interval data. Comput Stat Data Anal 52(3):1500–1515

Lin T (2009) Granular computing: practices, theories, and future directions. Encyclopedia of complexity and systems science. Springer, New York

Ljung L (1988) System identification: theory for the user. Prentice-Hall, Englewood Cliffs

Lu W, Chen X, Pedrycz W, Liu X, Yang J (2015) Using interval information granules to improve forecasting in fuzzy time series. Int J Approx Reason 57:1–18

Luna I, Ballini R (2012) Adaptive fuzzy system to forecast financial time series volatility. J Intell Fuzzy Syst 23(1):27–38

Luna I, Ballini R (2012) Online estimation of stochastic volatility for asset returns. In: IEEE computational intelligence for financial engineering and economics (CIFEr 2012), New York, NY, USA

Maciel L, Gomide F, Ballini R (2011) Evolving fuzzy systems for pricing fixed income options. In: IEEE workshop on evolving and adaptive intelligent systems EAIS, Paris, France, vol 1, pp 54–61

Maciel L, Gomide F, Ballini R (2012) MIMO evolving functional fuzzy models for interest rate forecasting. In: IEEE computational intelligence for financial engineering and economics (CIFEr 2012), New York, NY, USA

Maciel L, Gomide F, Ballini R (2014) Enhanced evolving participatory learning fuzzy modeling: an application for asset returns volatility forecasting. Evolv Syst 5(1):75–88

Maciel L, Gomide F, Ballini R (2015) Evolving possibilistic fuzzy modeling for financial interval time series forecasting. In: Proceedings of the the annual conference of the North American Fuzzy Information Processing Society (NAFIPS 2015), Redmond, WA, USA

McDonald S, Angelov P (2010) Evolving Takagi Sugeno modeling with memory for slow process. Int J Knowl-Based Intell Eng Syst 14(1):11–19

Moore RE, Kearfott RB, Cloud MJ (2009) Introduction to interval analysis. SIAM Press, Philadelphia

Narendra K, Parthasarathy K (1990) Identification and control of dynamical systems using neural networks. IEEE Trans Neural Netw 1(1):4–27

Pal NR, Pal K, Keller JM, Bezdek JC (2005) A possibilistic fuzzy c-means clustering algorithm. IEEE Trans Fuzzy Syst 13(4):517–530

Pedrycz W (2007) Granular computing—the emerging paradigm. J Uncertain Syst 1:38–61

Peters G (2011) Granular box regression. IEEE Trans Fuzzy Syst 19:1141–1152

Pettenuzzo D, Timmermann A, Valkanov R (2014) Forecasting stock returns under economic constraints. J Fin Econ 144(3):517–553

Rodrigues PMM, Salish N (2014) Modeling and forecasting interval time series with threshold models. In: Advances in data analysis and classification

Wang L, Liu X, Pedrycz W (2013) Effective intervals determined by information granules to improve forecasting in fuzzy time series. Expert Syst Appl 40(14):5673–5679

Weron R (2014) Electricity price forecasting: a review of the state-of-the-art with a look into the future. Int J Forecast 30(4):1030–1081

Xiong T, Bao Y, Hu Z (2014) Multiple-output support vector regression with a firefly algorithm for interval-valued stock price index forecasting. Knowl-Based Syst 55:87–100

Xiong T, Bao Y, Hu Z, Chiong R (2015) Forecasting interval time series using a fully complex-valued rbf neural network with dpso and pso algorithms. Inf Sci 305:77–92

Yang W, Han A, Wang S (2014) Forecasting financial volatility with interval-valued time series data. In: Vulnerability, uncertainty, and risk, pp 1124–1233

Yao J (2007) A ten-year review of granular computing. In: Proceedings of the 2007 IEEE international conference on granular computing, Silicon Valley, USA, pp 734–739

Zadeh L (1997) Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. Fuzzy Sets Syst 90(2):111–127