Evolving Possibilistic Fuzzy Modeling for Realized Volatility Forecasting with Jumps

Leandro Maciel, Rosangela Ballini, and Fernando Gomide

Abstract—Equity assets volatility modeling and forecasting provide key information for risk management, portfolio construction, financial decision making and derivatives pricing. Realized volatility models outperform autoregressive conditional heteroskedasticity and stochastic volatility models in out-of-sample forecasting. Gain in forecasting performance is achieved when models comprise volatility jump components. This paper suggests evolving possibilistic fuzzy modeling to forecast realized volatility with jumps. The modeling approach is based on an extension of the possibilistic fuzzy c-means clustering and on functional fuzzy rule-based models. It employs memberships and typicalities to recursively update cluster centers. The evolving nature of the model allows adding or removing clusters using statistical distance-like criteria to update the model as dictated by input data. The possibilistic model improves robustness to noisy data and outliers, an essential requirement in financial markets volatility modeling and forecasting. Computational experiments and statistical analysis are done using Value-at-Risk estimates to evaluate and to compare the performance of the evolving possibilistic fuzzy modeling with the Heterogeneous Autoregressive, neural networks models and current state of the art evolving fuzzy methods. The experiments use actual data from S&P 500 and Nasdaq (United States), FTSE (United Kingdom), DAX (Germany), IBEX (Spain) and Ibovespa (Brazil), major equity market indexes in global markets. The results show that the evolving possibilistic fuzzy model is highly efficient to model realized volatility with jumps in terms of forecasting accuracy.

Index Terms—Evolving fuzzy systems, possibilistic clustering, volatility forecasting, risk management.

I. INTRODUCTION

In finance and economics, volatility modeling and forecasting play an important role in derivative securities pricing, portfolio and risk management, investment analysis, and hedge operations [1]. Because the volatility of financial markets directly influences policymaking, volatility forecast is used to evaluate the vulnerability of financial markets and the economy.

Several studies suggest the use of daily returns to forecast daily volatility and the daily squared returns as a measure of true volatility [2]. Daily squared returns are found from closing prices and do not capture price fluctuations during the day. To alleviate this limitation, Andersen and Bollerslev [3] emphasize realized volatility as a measure of true volatility. Realized volatility is the sum of the squared high-frequency returns within a day. It conveniently avoids data analysis complications while covering more information during daily transactions.

The growing availability of financial market data at intraday frequencies has led to the development of improved volatility measurements [2]. Current literature advocates the use of realized variation measures in asset return volatility forecasting because they improve gains when compared against GARCH-family models and related stochastic volatility models [4], [5]. Further studies have revealed the importance of explicitly allowing for jumps in the estimation of volatility in the pricing of options and other derivatives [6].

Despite the evidence of more accurate performance of realized volatility models with jumps, traditional econometric approaches adopt restrictive assumptions that are not observable in practice. This is because of the complexity of the noisy, nonlinear, nonstationary dynamical behavior of volatility. Financial market data are affected by news, expectations, and psychological state of investors and market players. This induces volatility, adds noise, and causes outliers in data. Robustness is a key requirement when modeling systems subject to noisy data and outliers, as are financial time series.

Recently, [7] suggested a recursive possibilistic fuzzy c-means modeling (rPFM) approach for fuzzy rule-based in the form of Takagi-Sugeno (TS) fuzzy models. The rPFM uses a recursive form of the possibilistic fuzzy c-means (PFCM) [8] clustering algorithm to update the model structure, and employs the weighted recursive least squares algorithm to estimate the parameters of affine functions of the rule consequents. PFCM algorithm simultaneously produces memberships and typicalities through a hybridization of the possibilistic c-means (PCM) [9] and the fuzzy c-means (FCM) [10]. Typicalities alleviates outliers and noisy data sensitivity. Despite the potential of the rPFM modeling approach to deal with noisy data and outliers, it assumes that the model structure, i.e. the number of clusters, is known in advance. This is a major limitation in adaptive system modeling, especially when handling nonstationary data. To avoid this limitation, [12] suggests an

The FCM algorithm is sensitive to outliers and noise because it constraints the membership degrees to add up unity. For instance, considering two clusters, an outlier equidistant from the two cluster centers will have 0.5 membership degree, regardless of the absolute value of the distance of the outlier to the two cluster centers. The problem this creates is that outliers or noisy data which are far, but equidistant from the central structure of the two clusters, can nonetheless be given equal membership in both, when it seems more natural that such data be given very low, or even no membership in either cluster [8]. The possibilistic c-means (PCM) [9] avoids this problem relaxing the unity membership degrees sum constraint of FCM and view membership degrees as typicalities. PCM is very sensitive to initialization, and may produce coincident clusters [11]. Typicalities are susceptible to the choice of the additional parameters needed by the PCM as well.
evolving approach for possibilistic fuzzy c-means modeling using participatory learning to adapt the cluster structure and an utility measure to evaluate the quality of the current cluster structure. The model extends rPFM to deal with streams of data adaptively.

This paper suggests an evolving possibilistic fuzzy modeling approach (ePFM) for realized volatility forecasting with jumps. The ePFM adapts its structure with a possibilistic extension of the evolving Gustafson-Kessel-Like algorithm (eGKL) introduced in [13]. Contrary to [13], in this paper ePFM creates new clusters using a statistical control distance-based criteria and the cluster structure update mechanism uses both, memberships and typicalities. The ePFM modeling approach incorporates the advantages of the Gustafson-Kessel (GK) clustering algorithm of identifying clusters with different shape and orientation while recursively processing data. ePFM also uses an utility measure to evaluate the quality of the current cluster structure. The utility measure allows the rule base to shrink by removing rules with low utility (the data pattern shifted away from the domain of the rule) and produces parsimonious and more relevant rule base. The ePFM constitutes an alternative methodology for the construction of evolving possibilistic fuzzy models. Differently from [12], ePFM reduces the number of control parameters tuning the algorithm more autonomous and offers a new approach to update clusters.

The development of the ePFM approach targets realized volatility forecasting with jumps. In this context, possibilistic modeling helps to attenuate the effect of noise and outliers, adding robustness in financial volatility forecasting models, as the computational experiments show. The experiments use data of 1-minute returns from December 2009 through July 2013 for the main equity market indexes of major global markets such as S&P 500 and Nasdaq (United States), FTSE (United Kingdom), DAX (Germany), IBEX (Spain) and Ibovespa (Brazil). The performance of ePFM is evaluated and compared with the Heterogeneous Autoregressive model for realized volatility with jumps of [14], artificial neural networks approaches, and with current state of the art evolving fuzzy modeling algorithms. The results are further extended for risk management as an application for Value-at-Risk estimation. In this case, the capability of ePFM to handle nonlinear and time-varying dynamics such as assets returns volatility is essential, especially in real-time decision making and risk management.

Traditional volatility forecasting econometric benchmarks show several limitations e.g. i) linear structure; ii) restrictive assumptions about data distribution; iii) neglects uncertainties that induce data dynamics; iv) do not process steaming data; v) are affected by noisy data and outliers. The motivation of ePFM is to overcome these limitations by offering a more appropriate modeling approach for volatility dynamics and a nonlinear method for volatility forecasting with jumps. Summing up, the contributions of the paper include the following:

- new evolving framework for possibilistic fuzzy functional modeling;
- nonlinear approach for volatility forecasting with jumps;
- effective robust method to handle noisy data and outliers in volatility dynamics modeling;
- recursive, incremental method to process stream data in risk management;
- approach capable to access volatility clustering due to its clustering-based nature;
- use of actual data from developed and emergent economies to show its usefulness;
- economic evaluation of the results through VaR estimation.

Two recent approaches [15], [16] create, remove and update clusters of a cluster structure using a distance criterion. They assume that clusters are spherical and use deterministic decision rules to add, delete and update clusters. Contrary to [15], [16], the approach of this paper identifies clusters of different forms or sizes, and uses statistical information to adapt the cluster structure, namely, to add, remove or update the parameters of existing clusters.

The paper is organized as follows. A brief literature review on volatility forecasting with jumps models and evolving fuzzy systems methodologies is given in Section II. Section III reviews the nonparametric approach for realized volatility with jumps. The structure of functional, Takagi-Sugeno (TS) fuzzy model and the possibilistic fuzzy c-means clustering algorithm are summarized in Section IV. Section V introduces the evolving possibilistic fuzzy modeling approach and details its algorithmic steps. Computational results are summarized in Section VI. Section VII concludes the paper and enumerates issues for further investigation.

II. LITERATURE REVIEW

This section presents a brief overview of the literature concerning the current state of the art approaches for realized volatility forecasting with jumps and evolving fuzzy system models.

A. Realized volatility with jumps models

With the growing availability of realized volatility measurements, recent studies have revealed the importance of explicitly allowing for jumps in the estimation of volatility [6]. For example, [17] brought up that S&P 500 VIX implied volatility index subsumes information relating past jump contributions to the total volatility, and that VIX reflects incremental information pertaining to future jump activity. More recently, [18] shows that jumps have a positive and mostly significant impact on future volatility. Empirical analysis on the S&P 500 index, individual stocks and US bond yields reveal that this consideration improves significantly the accuracy of volatility forecasts especially for periods following the occurrence of jumps [18].

A recent review [19] gives empirical evidence of the predictive content of realized measures of jump power variations. Evaluation using S&P 500 futures data and stocks in the Dow 30 shows that past large jump power variations help much less in the prediction of future realized volatility than past small jump power variations [19].

The role of realized jumps detected from high frequency data in predicting future volatility from both statistical and economic perspectives is investigated in [20]. The authors...
state that a risk-averse investor can significantly improve the portfolio performance by incorporating realized jumps into a volatility timing based portfolio strategy.

Bipower variation is explored in nonparametric approaches that separate continuous sample path and jump components of realized volatility [21], [22]. Bipower variation is the summation of appropriately scaled cross-products of adjacent high-frequency absolute returns. The continuous sample path and jump components play different roles and, when taken into account separately, they increase forecasting performance significantly. The Heterogeneous Autoregressive (HAR) model departs from this framework [14] and uses lagged realized volatility sample paths and jumps as separate regressors for volatility. Using high-frequency prices from the foreign exchange, equity, and fixed-income markets, HAR shows the benefits of using jumps, and that jumps dynamics are much less persistent (and predictable) than the sample path dynamics.

Current literature emphasizes the potential of HAR modeling to deal with realized volatility forecasting with jumps against alternative approaches [23], [24], [25] and [26]. The papers examine whether nonlinear models, like principal components, neural networks, and GARCH are more accurate than HAR in realized volatility forecasting. The conclusion is that these models perform close to HAR, and that the realized volatility property of persistence is very important in realized volatility modeling and forecasting. HAR is an accurate realized volatility with jumps forecaster, but relies on linear regression of lagged sample paths and jumps that are not observable in practice.

Recent studies have indicated the use of evolving fuzzy and neural-fuzzy approaches to model and to forecast realized volatility of financial asset returns. These are nonlinear approaches that use lagged sample path and jump components from high-frequency data. For instance, [27] and [28] use a cloud-based evolving fuzzy model (eCloud) and a hybrid evolving neural fuzzy networks (eHFN), respectively, for volatility forecasting with jumps, and show that their accuracy are higher than traditional econometric techniques. Notably, the cloud-based evolving and neural fuzzy approaches embed a clustering mechanism that naturally captures volatility clustering stylized fact of financial time series. These models are compared against ePFM in this study.

B. Evolving fuzzy system models

Evolving fuzzy systems use incremental machine learning to process stream data and simultaneously adapt the structure and parameters of fuzzy models. In practice, linguistic and functional fuzzy rule-based (FRB) models are amongst the most prominent fuzzy models[29]. Functional evolving FRB models are adaptive, incremental fuzzy models in which the model structure (number of rules) and its parameters (membership function coefficients of the rule antecedents, coefficients of the rule consequents functions) are continuously evaluated as data are input. The model structure is found using an adaptive fuzzy clustering algorithm which may create, merge, and exclude clusters recursively. The coefficients of the consequent function is found using a recursive least squares (RLS) algorithm.

Examples of the different types of evolving fuzzy rule-based and fuzzy neural modeling approaches include the pioneering evolving Takagi-Sugeno (eTS) [30] approach and extensions (e.g. Simpl_eTS [31], xTended eTS (xtS) [32]). An autonomous user-free control parameters modeling scheme called eTS+ is given in [33]. The eTS+ uses criteria such as age, utility, local density, and zon of influence to update the model structure. Later, ePL+ was developed in the realm of participatory learning clustering [34]. ePL+ extends the ePL approach [35] and uses the updating strategy of eTS+.

An alternative method for evolving TS modeling is given in [36] based on a recursive form of the fuzzy c-means (rFCM) algorithm. Clustering aims at learning the model structure. Later, the rFCM method was translated in a recursive Gustafson-Kessel (rGK) algorithm. Similarly as the original off-line GK, the purpose of rGK is to capture different cluster shapes [37]. Combination of the rGK algorithm and evolving mechanisms such as adding, removing, splitting, merging clusters, and recursive least squares became a powerful evolving fuzzy modeling approach called eFuMo [38].

A distinct, but conceptually similar approach for TS modeling is the dynamic evolving neural fuzzy inference system model (DENFIS) [39]. DENFIS uses distance-based recursive clustering to adapt the rule base structure. The weighted recursive least squares with forgetting factor algorithm updates the parameters of rule consequents. A recursive clustering algorithm derived from a modification of the vector quantization technique, called evolving vector quantization, is another effective way to construct flexible fuzzy inference systems (FLEXFIS) [40].

More recent approaches for evolving fuzzy modeling include, to mention a few, the generic evolving neuro-fuzzy inference system (GENEFIS) [44], the parsimonious network based on fuzzy inference system (PANFIS) [45], the generalized smart evolving fuzzy system [46], the evolving fuzzy-rule-based parsimonious classifier (pClass) [47], and the meta-cognitive-based scaffolding recurrent classifier (rClass) [48]. Other important instances of evolving mechanisms are addressed in [41], [42]. A comprehensive overview of the evolving approaches can be found in [43].

III. REALIZED VOLATILITY WITH JUMPS

Consider a continuous-time jump diffusion process, expressed by the stochastic differential equation

$$dP(k) = \mu(k)dk + \sigma(k)dW(k) + \kappa_\sigma(k)d\kappa_\sigma(k),$$

with \(0 \leq k \leq T\), where \(P(k)\) is the logarithm of the asset price at time \(k\); \(\mu(k)\) is the continuous and locally bounded variation process; \(\sigma(k)\) is a strictly positive stochastic volatility process with a right continuous sample path, well-defined left limits, which allows for occasional jumps in volatility; \(W(k)\) is the standard Brownian motion; \(q(k)\) is the counting process with time-varying intensity \(\lambda(k)\), that is, \(\text{Prob}[d\kappa_\sigma(k) = 1] = \lambda(k)dk\); and \(\kappa_\sigma(k)\) corresponds to the size of the discrete
jumps in the price logarithm process. The quadratic variation for the cumulative return process, \( r(k) = P(k) - P(0) \), is
\[
r(k, 0) = \int_{0}^{k} \sigma^2(s)ds + \sum_{0 < \tau \leq k} \kappa^2_{\sigma}(s).
\]

(2)

In this case, the summation consists of the \( q(k) \) squared jumps that occur between time 0 and time \( k \). Recent literature emphasizes the importance to explicitly incorporate jumps in the price process e.g. [49], [50]. This paper adopts the nonparametric approach to perform the volatility dynamics estimates. The approach is based on the theory of bipower variation [21], which distinguishes jump from nonjump movements, and assumes that high-frequency data is available.

The daily realized volatility, \( RV \), or variation, is the sum of the corresponding \( 1/\Delta \) high-frequency intraday squared returns:
\[
RV_{k+1}(\Delta) = \sum_{j=1}^{1/\Delta} r_{k+j, \Delta, \Delta},
\]

(3)

where \( r_{k, \Delta} = P(k) - P(k - \Delta) \) is the discrete sample of the \( \Delta \)-period return.

The theory of quadratic variation states that the realized variation converges uniformly in probability to the increment of the quadratic variation process as the sampling frequency of the underlying returns increases [1], [14]:
\[
RV_{k+1}(\Delta) \rightarrow \int_{k}^{k+1} \sigma^2(s)ds + \sum_{k < s \leq k+1} \kappa^2_{\sigma}(s),
\]

(4)

for \( \Delta \rightarrow 0 \).

Realized volatility inherits the dynamics of both, the continuous sample path, and the jump processes. One can directly model and forecast realized volatility without distinguishing jumps or applying nonjump contributions. However, superior forecasts are achieved measuring and modeling the two components of realized volatility separately [14]. This paper follows the nonparametric framework and identifies the two components of the quadratic volatility process based on the concept of the standardized realized bipower variation BV theory [22]:
\[
BV_{k+1}(\Delta) = \mu_1^{-1} \sum_{j=2}^{1/\Delta} |r_{k+j, \Delta, \Delta}||r_{k+(j-1), \Delta, \Delta}|,
\]

(5)

where \( \mu_1 = \sqrt{(2/\pi)} = E(|Z|) \) is the mean of the absolute value of the standard normal distributed random variable, \( Z \). According to [22], as \( \Delta \rightarrow 0 \),
\[
BV_{k+1}(\Delta) \rightarrow \int_{k}^{k+1} \sigma^2(s)ds.
\]

(6)

As [22] shows, from (4) and (6), the contribution to the quadratic variation process due to the jumps (discontinuities) can be consistently estimated as \( \Delta \rightarrow 0 \) by:
\[
RV_{k+1}(\Delta) - BV_{k+1}(\Delta) \rightarrow \sum_{k < s \leq k+1} \kappa^2_{\sigma}(s).
\]

(7)

To avoid negative values of the right side of Equation (7) for a given finite sample, the measurements are truncated. Truncation ensures that all daily estimates are nonnegative [21]:
\[
J_{k+1}(\Delta) = \max [RV_{k+1}(\Delta) - BV_{k+1}(\Delta), 0].
\]

(8)

This paper adopts the Heterogeneous Autoregressive (HAR) approach [14] for the realized volatility formulation as benchmark. HAR is based on an extension of the heterogeneous ARCH (HARCH), models studied in [51]. The conditional variance is parametrized as a linear function of the lagged squared returns over the identical return horizon, together with the squared returns over longer and/or shorter return horizons [14]. The HAR model assumes multi period normalized realized variation
\[
RV_{k,k+h} = \frac{1}{h} \sum_{i=1}^{h} RV_{k,i}.
\]

(9)

The HAR model of [14] includes jump (\( J \)) components as follows:
\[
RV_{k+1} = \beta_0 + \beta_J RV_{k} + \beta_W RV_{k-5,k} + \beta_M RV_{k-22,k} + \beta_J J_{k+1},
\]

(10)

where \( h = 5 \) and \( h = 22 \) refer to the normalized measures of weekly and monthly volatilities, respectively, and \( \varepsilon \) is a white noise process, and \( k = 1,2,\ldots,T \).

HAR uses linear regression of past realized volatilities and jump components. Usually, the values of \( \beta \) are obtained via ordinary least squares. In practice, the volatility of asset returns exhibits a nonstationary, nonlinear dynamical behavior. This paper uses the evolving possibilistic fuzzy model (ePFM) for realized volatility forecasting with jumps instead of linear regression. Volatility clustering is one of the stylized facts of volatility. In volatility clustering, observations are cataloged by similarity. The ePFM modeling approach clusters data according to their common properties to form clusters and the corresponding fuzzy rule-based structure. Thus, ePFM naturally captures volatility clustering.

IV. TS MODEL AND POSSIBILISTIC FUZZY C-MEANS
This section introduces the notation and briefly reminds the basic constructs of the functional Takagi-Sugeno model and the possibilistic fuzzy c-means clustering algorithm.

A. Takagi-Sugeno fuzzy model
Takagi-Sugeno (TS) fuzzy model with affine consequents consists of a set of fuzzy functional rules of the form:
\[
R_i: \text{IF } x \text{ is } A_i \text{ THEN } y_i = \theta_{i0} + \theta_{i1}x_1 + \ldots + \theta_{im}x_m,
\]

(11)

where \( x = [x_1, x_2, \ldots, x_m]^T \in \mathbb{R}^m \) is the input, \( R_i \) is the \( i \)-th fuzzy rule, \( i = 1, 2, \ldots, c \), \( c \) is the number of fuzzy rules, \( A_i \) is the fuzzy set of the antecedent of the \( i \)-th fuzzy rule with membership function \( \mu_A(x): \mathbb{R}^m \rightarrow [0,1] \), \( y_i \in \mathbb{R} \) is the output of the \( i \)-th rule, and \( \theta_{i0} \) and \( \theta_{ij} \), \( j = 1, \ldots, m \), are the parameters of the consequent of the \( i \)-th rule.
The output of a TS model (11) is found as follows:

$$y = \frac{c}{\sum_{j=1}^{c} A_j(x) y_i} $$ \tag{12}

The expression (12) can be rewritten using normalized degrees of activation:

$$y = \sum_{i=1}^{c} \lambda_i y_i = \sum_{i=1}^{c} \lambda_i x_i^T \theta_i,$$ \tag{13}

where

$$\lambda_i = \frac{A_i(x)}{\sum_{j=1}^{c} A_j(x)},$$ \tag{14}

is the normalized firing level of the \(i\)-th rule, \(\theta_i = [\theta_{i1}, \ldots, \theta_{im}]^T\) is the vector of parameters, and \(x_i = [1, x_i^T]^T\) is the expanded input vector.

TS modeling requires learning of rules antecedents and consequents. Antecedents are found using fuzzy clustering algorithms, and parameters of the consequent function computed using least squares [29].

### B. Possibilistic fuzzy c-means clustering

The possibilistic fuzzy c-means clustering algorithm [8] can be summarized as follows. Let \(x_k = [x_{1k}, x_{2k}, \ldots, x_{mk}]^T \in \mathbb{R}^m\) be the input data at \(k\). A set of \(n\) inputs is denoted by \(X = \{x_k, k = 1, \ldots, n\}\), \(X \subset \mathbb{R}^{m \times n}\). The aim is to partition the data set \(X\) into \(c\) subsets called clusters.

A possibilistic fuzzy partition of the set \(X\) is a family \(\{A_i, 1 \leq i \leq c\}\). Each \(A_i\) is characterized by membership degrees and typicalities specified by the fuzzy and typicality partition matrices \(U = [u_{ik}] \in \mathbb{R}^{c \times n}\) and \(T = [t_{ik}] \in \mathbb{R}^{c \times n}\), respectively. The entries of the \(i\)-th row of matrix \(U\) is the values of membership (typicality) degrees of the data point in \(A_i\), \(V = [v_1, v_2, \ldots, v_c]^T \in \mathbb{R}^{c \times m}\) is the matrix of cluster centers.

The possibilistic fuzzy c-means (PFCM) clustering algorithm produces \(c\) vectors \(v_1, v_2, \ldots, v_c\) of \(c\) cluster centers, and fuzzy and typicality partition matrices \(U = [u_{ik}]\) and \(T = [t_{ik}]\) as a solution of the following optimization problem:

$$\min_{U,T,V} \left\{ \sum_{k=1}^{c} \sum_{i=1}^{n} (au_{ik})^{\eta_u} + bt_{ik}^{\eta_p} \right\},$$ \tag{15}

subject to \(\sum_{i=1}^{c} u_{ik} = 1 \forall k \) and \(0 \leq u_{ik}, t_{ik} \leq 1\).

Here \(a > 0, b > 0, \eta_u > 1, \eta_p > 1, \gamma_i > 0\) are chosen by the user, and \(D_{ki}^2\) is the distance between \(x_k\) and the \(i\)-th cluster centroid \(v_i\). The constants \(a\) and \(b\) weight the relative importance of fuzzy membership and typicality values in the objective function, and \(\eta_u\) and \(\eta_p\) are parameters associated with membership degrees and typicalities whose default values are \(\gamma_u = \eta_p = 2\).

Let \(V = [v_1, v_2, \ldots, v_c]^T \in \mathbb{R}^{c \times m}\) be the matrix of cluster centers. If \(D_{ki}^2 > 0\) for all \(i\), and \(X\) contains at least \(c\) distinct data points, then \((U, T, V) \in M_f \times M_p \times \mathbb{R}^{c \times m}\) minimizes \(J\), with \(1 \leq i \leq c\) and \(1 \leq k \leq n\), only if [8]:

$$u_{ik} = \left( \frac{c}{\sum_{j=1}^{c} (D_{ik} / D_{jk})^{2/(\eta_u-1)}} \right)^{-1},$$ \tag{16}

$$t_{ik} = \frac{1}{1 + \left( \frac{b}{\gamma_i} D_{ik}^2 \right)^{1/(\eta_p-1)}},$$ \tag{17}

$$v_i = \frac{\sum_{k=1}^{n} (au_{ik}^{\eta_u} + bt_{ik}^{\eta_p}) x_k}{\sum_{k=1}^{n} (au_{ik}^{\eta_u} + bt_{ik}^{\eta_p})},$$ \tag{18}

where

\(M_p = \{T \in \mathbb{R}^{c \times n} : 0 \leq t_{ik} \leq 1, \forall i, k; \forall k \ni it_{ik} > 0\}\),

\(M_f = \{U \in M_p : \sum_{k=1}^{c} u_{ik} = 1 \forall k; \sum_{k=1}^{n} u_{ik} > 0 \forall i\},\)

denote the set of possibilistic and fuzzy partition matrices, respectively.

Parameters \(\gamma_i\) are selected as follows[8]:

$$\gamma_i = K \sum_{k=1}^{n} u_{ik}^{\eta_u} D_{ik}^2, \quad 1 \leq i \leq c,$$ \tag{21}

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

### V. EVOLVING POSSIBILISTIC FUZZY MODELING

The evolving possibilistic fuzzy modeling approach (ePFM) is an extension of the evolving Gustafson-Kessel-Like clustering algorithm (eGKL) introduced in [13]. The ePFM considers membership and typicalities to update the cluster structure, i.e. the antecedents of the TS fuzzy rule-based model. The ePFM includes a utility measure to avoid unused clusters that persists during learning.

#### A. Antecedents learning

The ePFM uses the objective function of possibilistic fuzzy clustering algorithm as in (15). The distance \(D_{ki}^2\) is the Mahalanobis distance, which is a squared inner-product distance norm that depends on a positive definite symmetric matrix \(A_{ik}\) as follows:

$$D_{ki}^2 = \|x_k - v_i\|_{A_{ik}}^2 = (x_k - v_i)^T A_{ik} (x_k - v_i).$$ \tag{22}

The matrix \(A_{ik}\), \(i = 1, \ldots, c\), determines the shape and orientation of the cluster \(i\), and is computed from estimates of the data dispersion:

$$A_{ik} = [\rho_i \det(F_{ik})]^{1/m} F_{ik}^{-1},$$ \tag{23}

where \(\rho_i\) is the cluster volume of the \(i\)-th cluster (usually \(\rho_i = 1\) for all clusters) and \(F_{ik}\) is the fuzzy dispersion matrix:

$$F_{ik} = \frac{\sum_{k=1}^{n} u_{ik}^{\eta_u} (x_k - v_i)(x_k - v_i)^T}{\sum_{k=1}^{n} u_{ik}^{\eta_p} A_{ik}}.$$ \tag{24}

Most of the fuzzy clustering algorithms assume clusters with spherical shapes. Actually, in real world applications clusters often have different shapes and orientations in the data space. A way to distinguish cluster shapes is to use information about the dispersion of the input data as the Mahalanobis distance does.

Rules antecedents learning with ePFM relies on recursive possibilistic fuzzy clustering algorithm with the Mahalanobis distance. Evolving mechanisms to create and to update clusters.
come from the adaptive principles of [13] which in turn is inspired by the recursive the k-nearest neighbor (kNN) [52] and the linear vector quantization (LVQ) [53].

Assume an input data \( x_k \) at step \( k \). If \( x_k \) falls within an existing cluster, then the corresponding cluster center is updated. Otherwise, the input data may define a new cluster center. More specifically, suppose a cluster structure with \( c \) clusters when the \( k \)-th data is input. The similarity between the input data \( x_k \) with each of the existing \( c \) clusters is evaluated using the Mahalanobis distance (22). Consider the following condition:

\[
D^2_{ik} < \chi^2_{m, \beta}, \quad i = 1, \ldots, c,
\]

where \( \chi^2_{m, \beta} \) is the \((1 - \beta)\)-th value of the chi-squared distribution with \( m \) degrees of freedom and \( \beta \) is the probability of false alarm.\(^3\)

Inequality (25) is motivated by statistical process control to identify variations in systems that are due to actual input changes rather than process noise. See [13] for details.

If condition (25) holds, then clusters remain the same and the minimum value of \( D_{ik} \) determines the closest cluster \( p \):

\[
p = \operatorname{arg\,min}_{i=1, \ldots, c}(D_{ik}), \quad D^2_{ik} < \chi^2_{m, \beta}, \quad i = 1, \ldots, c.
\]

Input \( x_k \) is assigned to the \( p \)-th cluster and

\[
M_{p,\text{new}} = M_{p,\text{old}} + 1,
\]

where \( M_i \) counts the number of data points that fall within the boundary of cluster \( i, \quad i = 1, \ldots, c \).

The eGKL algorithm of [13] updates the \( p \)-th cluster center using the Kohonen-like rule [53]:

\[
v_{p,\text{new}} = v_{p,\text{old}} + \alpha(x_k - v_{p,\text{old}}),
\]

where \( \alpha \) is a learning rate, and \( v_{p,\text{new}} \) and \( v_{p,\text{old}} \) denote the new and old values of the cluster center, respectively. Contrary to [13], the evolving possibilistic fuzzy modeling suggested in this paper updates the \( p \)-th cluster center proportionally to the gradient of the possibilistic fuzzy clustering objective function (15) with respect to \( v_i \). That is, the \( p \)-th cluster center is updated as follows:

\[
v_{p,\text{new}} = v_{p,\text{old}} + \alpha(\eta_{pk} + \beta_{pk})A_{pk}(x_k - v_{p,\text{old}}).
\]

The determinant and the inverse of the dispersion matrix of the \( p \)-th cluster is updated using\(^4\):

\[
F^{-1}_{p,\text{new}} = (I - G_p(x_k - v_{p,\text{old}}))F^{-1}_{p,\text{old}} \frac{1}{1 - \alpha},
\]

\[
\det(F_{p,\text{new}}) = (1 - \alpha)^{-m}\det(F_{p,\text{old}})(1 - \alpha + \alpha \cdot (x_k - v_{p,\text{old}})^T F^{-1}_{p,\text{old}} (x_k - v_{p,\text{old}})),
\]

where

\[
G_p = F^{-1}_{p,\text{old}}(x_k - v_{p,\text{old}})\alpha \cdot (1 - \alpha + \alpha(x_k - v_{p,\text{old}})^T F^{-1}_{p,\text{old}} (x_k - v_{p,\text{old}}))^{-1}.
\]

The remaining clusters centers are updated in the opposite direction to move them away from the \( p \)-th cluster:

\[
v_{q,\text{new}} = v_{q,\text{old}} - \alpha(\eta_{qk} + \beta_{qk})A_{qk}(x_k - v_{q,\text{old}}),
\]

where \( q = 1, \ldots, c, \quad q \neq p \).

The parameter \( M_i, \quad i = 1, \ldots, c \), gives a measure of the reliability of the estimated clusters. The minimal value \( M_{\min} \) is the minimal number of data points needed to learn the entries of the \( i \)-th inverse dispersion matrix \( F^{-1}_{i,k} \). \( M_{\min} \) can be estimated from the dimension \( m \) of the input data:

\[
M_{\min} = Qm(m + 1)/2,
\]

where \( Q \) is a reliability parameter, with default value \( Q = 2 \).

If (25) does not hold, then \( x_k \) is not close enough to any of the cluster centers, and the natural action is to create a new cluster. One must check whether this fact is not due to the lack of reliable clusters surrounding \( x_k \):

\[
M_p < M_{\min},
\]

\[
p = \operatorname{arg\,min}_{i=1, \ldots, c}(D_{ik}). \]

If (34) holds, then the closest cluster \( p \) is updated using (29)-(31). Otherwise, a new cluster is created, \( c_{\text{new}} = c_{\text{old}} + 1 \) with the following initialization:

\[
v_{c,\text{new}} = x_k, \quad F^{-1}_{c,\text{new}} = F^{-1} = \kappa I,
\]

\[
\det(F_{c,\text{new}}) = \det(F_0), \quad M_{c,\text{new}} = 1,
\]

where \( I \) is an \((m \times m)\) identity matrix and \( \kappa \) is a sufficient large positive number. Initialization (35) is also adopted when there is no data are initially available.

B. Cluster quality measurement

The quality of the cluster structure at each step \( k \) is monitored considering the utility index introduced in [33]. The utility index is an indicator of the accumulated relative firing level of a rule:

\[
\mathcal{U}_i(k) = \frac{\sum_{l=1}^{k} \lambda_i}{K^{i*}},
\]

where \( K^{i*} \) is the step at which cluster \( i^{*} \) was created.

Thus, the utility index indicates how much the rule has been used since its creation. The idea is to use \( \mathcal{U}_i \) to avoid keeping unused clusters in the cluster structure. Clusters with low quality may be deleted. Originally, [33] suggests to delete a rule whenever \( \mathcal{U}_i \) is less or equal than an user specified threshold. The ePFM algorithm adopts a more flexible, user free approach; rule \( i \) is deleted

\[
\text{IF } \mathcal{U}_i(k) \leq \bar{\mathcal{U}}(i) - 2\sigma_{\mathcal{U}_i} \text{ THEN } c_{\text{new}} = c_{\text{old}} - 1,
\]

where \( \bar{\mathcal{U}} \) and \( \sigma_{\mathcal{U}_i} \) are the sample average and the standard deviation of the utility of cluster \( i \) data.

Condition (37) means that if the utility of cluster \( i \) at step \( k \) is less than or equal 2 standard deviation of the average utility of cluster \( i \), then cluster \( i \) has low utility and can be removed. This idea is similar to the 2\( \sigma \) process control band in statistical process control, considering the tail of the left side of utility distribution once it is associated with lower utility values.
C. Consequent parameters estimation

Estimation of the coefficients of the affine rule consequents is done using the weighted recursive least squares algorithm (wRLS) [54] as in [33], [46], [55]. Briefly, expression (13) can be rewritten as:

\[ y = \Lambda^T \Theta_i \]

where \( \Lambda = [\lambda_1 x_1^T, \lambda_2 x_2^T, \ldots, \lambda_n x_n^T]^T \) is the fuzzily weighted extended input, \( x_i = [1 x_i^T]^T \) the expanded data vector, and \( \Theta = [\theta_{i1}, \theta_{i2}, \ldots, \theta_{ic}]^T \) the parameter matrix, \( \theta_i = [\theta_{i0}, \theta_{i1}, \ldots, \theta_{im}]^T \).

Assuming that the actual outputs can be obtained at each step, the coefficients of the consequents can be updated using the recursive least squares (RLS) algorithm considering either local or global optimization. This paper uses the local wRLS:

\[ \min_{\theta_i} F^T_{\text{WL}} = \min_{\theta_i} \sum_{k=1}^{n} \left( y_{ik} - x_{\text{ck}}^T \theta_i \right)^2. \]  

The recursive calculation of \( \theta_i \) using (39) results in the wRLS [54] equations:

\[ \theta_{i,k+1} = \theta_{i,k} + \sum_{i,k} x_{\text{ck}} \lambda_{ik} (y_{ik} - x_{\text{ck}}^T \theta_{i,k}), \quad \theta_{i,0} = 0, \]
\[ \sum_{i,k} - \lambda_{ik} \sum_{i,k} x_{i,k} x_{\text{ck}}^T \Sigma_{i,k} \sum_{i,k} + 1 = \lambda_{ik} x_{i,k} x_{\text{ck}}^T \Sigma_{i,k} \Sigma_{i,k} + \Omega I, \hspace{1cm} \Sigma_{i,k} = \Omega I, \]

where \( I \) is an \((m \times m)\) identity matrix, \( \Omega \) a large number (usually \( \Omega = 1000 \)), and \( \Sigma \) is the dispersion matrix.

If at step \( k \) a new fuzzy rule is added, \( c \leftarrow c + 1 \), and \( \Sigma_{c+1,k} \) are initialized as follows:

\[ \theta_{c+1,k} = 0, \]
\[ \Sigma_{c+1,k} = \Omega I. \]

D. ePFM algorithm

The evolving possibilistic fuzzy modeling (ePFM) approach is summarized below. The steps of the algorithm are non-iterative. The procedure may adapt the existing model as demanded by input data. Its recursive nature means that it can be used online and, as far as data storage is concerned, it is memory efficient.

<table>
<thead>
<tr>
<th>Evolving possibilistic fuzzy modeling</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Compute ( \gamma_i ) and ( \psi_{mi} ) as terminal FCM partitions. Let ( M_{\text{max}} = Q(m + 1)/2 ).</td>
</tr>
<tr>
<td>2. Choose control parameters ( a, b, \alpha, \beta, \kappa ), and set ( F_0^{-1} = \kappa I ).</td>
</tr>
<tr>
<td>3. for ( k = 1, 2, \ldots ) do</td>
</tr>
<tr>
<td>4. read input data ( x_i )</td>
</tr>
<tr>
<td>5. check the affinity of ( x_i ) with existing clusters: ( D_{i,k} &lt; \psi_{mi,\beta} ),</td>
</tr>
<tr>
<td>6. identify the closest cluster: ( p = \arg \min_{c=1,\ldots,n} (D_{ik}) )</td>
</tr>
<tr>
<td>7. if ( D_{ik} &lt; \psi_{mi,\beta} ) or ( M_k &lt; M_{\text{max}} ) then</td>
</tr>
<tr>
<td>8. update the parameters of the ( p )-th cluster using (29)-(31)</td>
</tr>
<tr>
<td>9. move away the centers of the remaining clusters using (32)</td>
</tr>
<tr>
<td>10. else</td>
</tr>
<tr>
<td>11. create a new cluster: ( c_{\text{new}} = c_{\text{add}} + 1 )</td>
</tr>
<tr>
<td>12. initialize the new cluster using (35)</td>
</tr>
<tr>
<td>13. end if</td>
</tr>
<tr>
<td>14. if ( D_{ik} \leq (b_k - 2\alpha_k) ) then delete cluster ( i )</td>
</tr>
<tr>
<td>15. ( c_{\text{new}} = c_{\text{add}} + 1 )</td>
</tr>
<tr>
<td>16. end if</td>
</tr>
<tr>
<td>17. compute rule consequent parameters using the wRLS</td>
</tr>
<tr>
<td>18. compute model output ( y_{k+1} )</td>
</tr>
<tr>
<td>19. end for</td>
</tr>
</tbody>
</table>

The ePFM has theoretical and computational advantages in realized volatility forecasting. Theoretically, because volatility clustering catalogs data by similarity, and because ePFM sets the number of clusters through an adaptive clustering algorithm, ePFM naturally captures volatility clustering whenever it develops. Computationally, the ePFM is a fully recursive algorithm, capable to perform online, which is a highly desirable characteristic when modeling high frequency, nonstationary, nonlinear dynamic systems such as current financial markets.

VI. COMPUTATIONAL EXPERIMENTS

The ePFM approach translates into a flexible modeling procedure that can be applied to a range of problems such as e.g. process modeling, time series forecasting, classification, system control, and novelty detection. Our aim in this paper to suggest and to evaluate the performance of ePFM for stock market realized volatility forecasting including jump components. The ePFM is a non-linear, adaptive, and robust approach for realized volatility modeling. The results produced by ePFM are compared with the HAR [14] benchmark, with traditional artificial neural networks such as MLP and ANFIS, and with the evolving fuzzy models xTS [56], eTS+ [33], eCloud [27], and eHFN [28]. Comparisons are made using Value-at-Risk (VaR) estimates. This section overviews the source of data, details the evaluation methodology adopted, presents the results, and discusses comparisons in terms of goodness of fit and VaR modeling.

A. Data

The data sets are from the main equity market indexes in the global markets: S&P 500 and Nasdaq (United States), FTSE (United Kingdom), DAX (Germany), IBEX (Spain) and IBOVESPA (Brazil). The data sets are useful to evaluate ePFM performance in developed and emergent economies. Data contain 1-minute quotations from December 7, 2009, through July 31, 2013, for a total of 485,364 intraday observations. The high-frequency spot quotations are from Bloomberg. The stock market data result in a total of \( 1/\Delta = 507 \) high-frequency return observations per day. Henceforth, for simplicity, reference to \( \Delta \) is omitted. This paper refers to the 1-minute realized volatility and jump measures as \( RV \) and \( J \), respectively.

B. Methodology

Nonparametric model of realized volatility [1] used (3) and 1-minute equity indexes data. From the bipower variation theory, the jump components of realized variation can be separated as in (6) and (8). The benchmark is the HAR model [14] (see Equation (10)).

The data set is divided into an in-sample, the data from December 7, 2009 through December 30, 2011, and an out-of-sample, composed of all the remaining data. The inputs of the neural network methods and the evolving fuzzy models are the same as the explanatory variables of the linear regression benchmark HAR \( (RV_k, RV_{k-5}, k, RV_{k-22}, k, J_k) \).

5 The sample considered is justified by the increase in high-frequency data available in this period.
The model output is the one-step ahead forecast of realized volatility, $RV_{k+1}$.  
Comparison of volatility forecasts assumes mean-squared forecast error (MSFE), mean absolute forecast error (MAFE), and mean percentage forecast error (MPFE):  

$$MSFE = \frac{1}{T} \sum_{i=1}^{T} (RV_i - \hat{RV}_i)^2,$$  

$$MAFE = \frac{1}{T} \sum_{i=1}^{T} |RV_i - \hat{RV}_i|,$$  

$$MPFE = \frac{1}{T} \sum_{i=1}^{T} \frac{|RV_i - \hat{RV}_i|}{RV_i},$$  


where $T$ is the number of out-of-sample observations, $RV_i$ is the actual realized volatility at period $i$, and $\hat{RV}_i$ is the realized volatility forecast at $i$.

This paper employs the Diebold–Mariano [57] statistic test to evaluate the null hypothesis of equal predictive accuracy. If $T$ is the sample size and $e_1^i$ and $e_2^i$ ($i = 1, 2, \ldots, T$) are the forecast errors of two competing forecasts, then the loss function is estimated as

$$L_1(e_i^1) = |e_1^i| \text{ and } L_2(e_i^2) = |e_2^i|.\text{ (47)}$$

The Diebold–Mariano test is based on the loss differential:

$$d_i = L_1(e_i^1) - L_2(e_i^2).\text{ (48)}$$

Using $d_i$, the null hypothesis $H_0$: $E(d_i) = 0$ is tested against the alternative $H_1$: $E(d_i) \neq 0$. The Diebold–Mariano test statistic, DM, is estimated by

$$DM = \frac{\bar{d}}{\sqrt{\bar{V}(\bar{d})}} \sim N(0, 1),\text{ (49)}$$

where

$$\bar{V}(\bar{d}) = T^{-1} \sum_{i=1}^{T} \{q_0 + 2 \sum_{k=1}^{T-1} \Phi_k\} - T^{-1} \sum_{j=1}^{T} d_i - \bar{d}(T - \bar{d})$$

and $\bar{d}$ is the sample mean of $d_i$.

The results are evaluated in terms of Value-at-Risk estimation. Value-at-Risk (VaR) has been adopted by practitioners and regulators as the standard mechanism to measure market risk of financial assets. It encapsulates in a single quantity the potential market value loss of a financial asset over a time horizon $h$, at a significance or coverage level $\alpha_{VaR}$. Alternatively, it reflects the asset market value loss over the time horizon $h$, that is not expected to be exceeded with probability $1 - \alpha_{VaR}$, i.e.:  

$$\text{Prob} (r_{k+h} \leq \text{VaR}_{k+h}) = 1 - \alpha_{VaR} \text{ (50)}$$

where

$$r_{k+h} = \frac{ln(P_{k+h})}{ln(P_k)}, \text{ (51)}$$

is the asset log return over the period $h$ and $P_j$ is the asset price at $j$.

Hence, VaR is the $\alpha_{VaR}$-th quantile of the conditional returns distribution defined as: $\text{VaR}_{k+h} = CDF^{-1}_{k+h}(\alpha_{VaR})$, where $CDF(\cdot)$ is the cumulative distribution function of the returns, and $CDF^{-1}(\cdot)$ denotes its inverse. Here, we concentrate at $h = 1$ as it bears the greatest practical interest.

Assume that the daily conditional heteroskedastic returns in (51) of a financial asset is described by the process:

$$r_k = RV_k z_k,\text{ (52)}$$

where $z_k \sim i.i.d(0, 1)$ and $RV_k$ is the asset realized volatility at $k$.

Thus, the VaR at $k + 1$ is given by:

$$\text{VaR}_{k+1} = RV_{k+1} CDF^{-1}_{k+1}(\alpha_{VaR}),\text{ (53)}$$

where $CDF^{-1}_{k+1}(\alpha_{VaR})$ is the critical value from the normal distribution table at $\alpha_{VaR}$ confidence level. This paper assumes $\alpha_{VaR} = 5\%$ confidence level for all models.

The performance is evaluated with VaR estimates using two loss functions: the violation ratio, and the average square magnitude function. The violation ratio (VR) is the percentage occurrence of an actual loss that is greater than the estimated maximum loss in the VaR framework. VR is computed as follows:

$$\text{VR} = \frac{1}{T} \sum_{k=1}^{T} \Phi_k,\text{ (54)}$$

where $\Phi_k = 1$ if $r_k < \text{VaR}_k$ and $\Phi_k = 0$ if $r_k \geq \text{VaR}_k$, where $\text{VaR}_k$ is the one step ahead forecasted VaR for day $k$, and $T$ is the number of observations in the test set.

The average square magnitude function (ASMF) [58] considers the amount of possible default measuring the average squared cost of exceptions. It is found using:

$$\text{ASMF} = \frac{1}{\vartheta} \sum_{j=1}^{\vartheta} \xi_j,\text{ (55)}$$

where $\vartheta$ is the number of exceptions of the respective model, $\xi_j = (r_j - \text{VaR}_j)^2$ when $r_j < \text{VaR}_j$ and $\xi_j = 0$ when $r_j \geq \text{VaR}_j$. The average squared magnitude function enables us to distinguish between models with similar or identical hit rates [59].

All modeling approaches are characterized in terms of the average number of rules/clusters/nodes and the (CPU) time needed to process test data. The algorithms were implemented and run using Matlab on a laptop equipped with 4 GB and Intel® i3CPU.

C. Results

Table I summarizes the basic statistics of S&P 500, Nasdaq, FTSE, DAX, IBEX and IBOVESPA indexes returns for the period from December 2009 to July 2013. The returns mean are very close to zero and the standard deviations are similar for all indexes. The skewness coefficient is negative for all stock indexes, which is in line with the stylized fact of heavy tails in financial time series, also confirmed by the high values of kurtosis. Moreover, the Jarque-Bera (JB) [60] statistics indicate that the return series are non-normal with a 99% confidence interval. As an illustration, figures 1-3 show the returns, realized volatility and jumps of S&P 500, DAX, and IBOVESPA indexes, respectively. All markets show volatility clusters, especially during more unstable periods, such as
during the 2008 subprime crisis in USA that affected most world economies. The jumps are mostly associated with the periods of higher volatility.

**TABLE I**

<table>
<thead>
<tr>
<th>Statistics</th>
<th>S&amp;P 500</th>
<th>Nasdaq</th>
<th>FTSE</th>
<th>DAX</th>
<th>IBEX</th>
<th>IBOVESPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.00049</td>
<td>0.00051</td>
<td>0.00062</td>
<td>0.00036</td>
<td>-0.00034</td>
<td>-0.00035</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>0.01711</td>
<td>0.02222</td>
<td>0.00862</td>
<td>0.01881</td>
<td>0.01728</td>
<td>0.01887</td>
</tr>
<tr>
<td>Skewness</td>
<td>0.26285</td>
<td>0.43000</td>
<td>-0.17089</td>
<td>-0.00113</td>
<td>-0.40905</td>
<td>-0.26510</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>7.03841</td>
<td>6.27430</td>
<td>4.92608</td>
<td>-3.46409</td>
<td>0.83406</td>
<td>2.05081</td>
</tr>
<tr>
<td>Maximum</td>
<td>0.04862</td>
<td>0.05199</td>
<td>0.05012</td>
<td>0.05337</td>
<td>0.12484</td>
<td>0.04974</td>
</tr>
<tr>
<td>Minimum</td>
<td>0.00096</td>
<td>0.01529</td>
<td>-0.03779</td>
<td>-0.06379</td>
<td>-0.06347</td>
<td>-0.00431</td>
</tr>
<tr>
<td>JB</td>
<td>645.03</td>
<td>425.68</td>
<td>143.89</td>
<td>234.58</td>
<td>903.48</td>
<td>171.04</td>
</tr>
<tr>
<td>p-value</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
</tr>
</tbody>
</table>

**Fig. 1.** S&P 500 daily returns (top panel), realized volatility (middle panel) and the jump component (bottom panel).

**Fig. 2.** DAX daily returns (top panel), realized volatility (middle panel) and the jump component (bottom panel).

**Fig. 3.** IBOVESPA daily returns (top panel), realized volatility (middle panel) and the jump component (bottom panel).

The control parameters of ePFM were chosen from experiments conducted to find the best performance in terms of the forecasting error. Table II shows the ePFM parameters for each of the equity market indexes considered in this paper. Initialization uses the FCM algorithm to choose $\gamma_i$ and $\nu_0$. One must note that is necessary to choose the probability of a false alarm $\beta$ and then calculate $\chi_{m, \beta}^2$. This work considers a default probability of false alarm $\beta = 0.0455$ that relates to the $2\sigma$ process control band in the single-variable statistical process control as [13]. Table III shows the $\chi_{m, 0.0455}^2$ for different values of $m$. The ePFM uses $\chi_{3, 0.0455}^2 = 9.7156$ for all indexes because the number of inputs is $m = 4$. Control parameters of the remaining evolving methods were chosen similarly as ePFM.

**TABLE II**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>S&amp;P 500</th>
<th>Nasdaq</th>
<th>FTSE</th>
<th>DAX</th>
<th>IBEX</th>
<th>IBOVESPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$b_1$</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>$\alpha_1$</td>
<td>0.06</td>
<td>0.09</td>
<td>0.08</td>
<td>0.10</td>
<td>0.09</td>
<td>0.12</td>
</tr>
<tr>
<td>$\kappa_1$</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>S&amp;P 500</th>
<th>Nasdaq</th>
<th>FTSE</th>
<th>DAX</th>
<th>IBEX</th>
<th>IBOVESPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_2$</td>
<td>2</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$b_2$</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>$\alpha_2$</td>
<td>0.06</td>
<td>0.09</td>
<td>0.08</td>
<td>0.10</td>
<td>0.09</td>
<td>0.12</td>
</tr>
<tr>
<td>$\kappa_2$</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>

**TABLE III**

| $\chi_{m, 0.0455}^2$ for different values of $m$. |
|------------------|------------------|------------------|------------------|------------------|------------------|
| $m$              | $\chi_{1, 0.0455}^2$ | $\chi_{2, 0.0455}^2$ | $\chi_{3, 0.0455}^2$ | $\chi_{4, 0.0455}^2$ |
| 1                | 1.0861           | 1.3605           | 1.7150           | 2.1085           |
| 2                | 2.8743           | 4.1592           | 5.4409           | 7.3151           |
| 3                | 7.0384           | 9.7156           | 12.704           | 17.011           |

Table IV summarizes the performance of the models in terms of mean squared forecast error (MSFE), mean average forecast error (MAFE) and mean percentage forecast error (MPFE). The results indicate the higher capability of the evolving possibilistic fuzzy modeling and the evolving fuzzy models in handling realized volatility forecasting with jumps. The HAR model gives the worst results because volatility clearly is nonlinear. The ePFM and the evolving fuzzy models present similar forecast error, mainly ePFM, eFHN and eCloud. The evolving possibilistic approach gives better fit, especially for MPFE. The neural network models, MLP and ANFIS, show higher accuracy than the HAR approach, however, the are inferior than the evolving methods in terms of the error metrics. Despite the distinct volatility dynamics, the predictive accuracy of the the equity market is similar. These results agree with [27] and [28] which suggest that clustering based volatility modeling and forecasting techniques perform better when handling volatility clustering.

Figures 4-9 depict the actual realized volatility and the estimates produced by ePFM approach for S&P 500, Nasdaq, FTSE, DAX, IBEX and IBOVESPA, respectively. Notice that ePFM captures the volatility dynamics despite the nature of the different economies.

In addition to goodness of fit, as mirrored by forecast error, the models were evaluated statistically. The Diebold–Mariano [57] test statistics are summarized in Table V. The test is performed for each pair of models. The null hypothesis of equal predictive accuracy of the the evolves against the econometric benchmark HAR only.

6The neural network-based models MLP and ANFIS achieve lower accuracy than the evolving fuzzy methods. Because the focus of this paper is on an evolving approach for volatility forecasting with jumps and risk management, henceforth the comparisons will highlight the evolving approaches against the econometric benchmark HAR only.
predictive accuracy is rejected with 5% confidence level, that is if $|DM| > 1.96$. From this point of view, the HAR model is statistically inferior in terms of accuracy to all remaining realized volatility forecasting algorithms for the equity market indexes data considered in this paper. Statistically, ePFM and the remaining evolving fuzzy models are equally accurate ($|DM| < 1.96$).

<table>
<thead>
<tr>
<th>Method</th>
<th>S&amp;P 500</th>
<th>Nasdaq</th>
<th>FTSE</th>
<th>DAX</th>
<th>IBEX</th>
<th>IBOV</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAR</td>
<td>0.8726</td>
<td>1.0918</td>
<td>1.4450</td>
<td>0.9971</td>
<td>1.0117</td>
<td>1.1382</td>
</tr>
<tr>
<td>MLP</td>
<td>0.0982</td>
<td>0.1102</td>
<td>0.2673</td>
<td>0.0730</td>
<td>0.0874</td>
<td>0.0652</td>
</tr>
<tr>
<td>ANFIS</td>
<td>0.0824</td>
<td>0.1098</td>
<td>0.1842</td>
<td>0.0662</td>
<td>0.0788</td>
<td>0.0663</td>
</tr>
<tr>
<td>xTS</td>
<td>3.0E-06</td>
<td>2.0E-05</td>
<td>4.0E-04</td>
<td>1.1E-06</td>
<td>2.6E-06</td>
<td>2.1E-05</td>
</tr>
<tr>
<td>eTS+</td>
<td>2.3E-07</td>
<td>3.6E-07</td>
<td>9.7E-06</td>
<td>4.7E-07</td>
<td>4.1E-05</td>
<td>2.1E-06</td>
</tr>
<tr>
<td>eCloud</td>
<td>3.2E-06</td>
<td>1.9E-06</td>
<td>2.9E-05</td>
<td>9.4E-06</td>
<td>2.5E-06</td>
<td>4.2E-06</td>
</tr>
<tr>
<td>eHFN</td>
<td>1.8E-07</td>
<td>1.3E-07</td>
<td>4.5E-08</td>
<td>3.8E-07</td>
<td>1.5E-06</td>
<td>5.0E-07</td>
</tr>
<tr>
<td>ePFM</td>
<td>1.3E-06</td>
<td>1.4E-05</td>
<td>2.8E-04</td>
<td>1.1E-06</td>
<td>1.6E-06</td>
<td>1.3E-05</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>HAR</th>
<th>MLP</th>
<th>ANFIS</th>
<th>xTS+</th>
<th>eTS+</th>
<th>eCloud</th>
<th>eHFN</th>
<th>ePFM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAFE</td>
<td>1.0092</td>
<td>0.7628</td>
<td>1.2422</td>
<td>1.0245</td>
<td>0.9887</td>
<td>1.1762</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0329</td>
<td>0.0189</td>
<td>0.0982</td>
<td>0.0455</td>
<td>0.0165</td>
<td>0.0770</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.0274</td>
<td>0.0166</td>
<td>0.0067</td>
<td>0.0413</td>
<td>0.0149</td>
<td>0.0690</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.1E-02</td>
<td>2.1E-02</td>
<td>1.0E-02</td>
<td>2.0E-02</td>
<td>1.5E-02</td>
<td>1.4E-03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.2E-03</td>
<td>8.5E-04</td>
<td>4.6E-04</td>
<td>9.8E-04</td>
<td>2.9E-04</td>
<td>2.1E-03</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.7E-04</td>
<td>2.9E-04</td>
<td>3.3E-04</td>
<td>1.4E-04</td>
<td>2.4E-04</td>
<td>2.5E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.7E-04</td>
<td>1.9E-04</td>
<td>1.2E-04</td>
<td>3.4E-04</td>
<td>6.7E-04</td>
<td>4.3E-04</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.1E-03</td>
<td>1.5E-03</td>
<td>1.7E-03</td>
<td>1.4E-03</td>
<td>2.9E-03</td>
<td>1.2E-02</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>HAR</th>
<th>MLP</th>
<th>ANFIS</th>
<th>xTS+</th>
<th>eTS+</th>
<th>eCloud</th>
<th>eHFN</th>
<th>ePFM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4.365</td>
<td>4.656</td>
<td>7.763</td>
<td>4.991</td>
<td>4.324</td>
<td>4.110</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.009</td>
<td>4.201</td>
<td>6.998</td>
<td>4.720</td>
<td>3.800</td>
<td>3.836</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.833</td>
<td>2.902</td>
<td>5.452</td>
<td>2.783</td>
<td>2.374</td>
<td>1.872</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.534</td>
<td>2.398</td>
<td>5.251</td>
<td>2.482</td>
<td>2.191</td>
<td>2.011</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.923</td>
<td>2.510</td>
<td>4.387</td>
<td>2.333</td>
<td>1.801</td>
<td>1.672</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.348</td>
<td>2.438</td>
<td>4.983</td>
<td>2.516</td>
<td>1.953</td>
<td>1.759</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.752</td>
<td>1.874</td>
<td>4.117</td>
<td>2.134</td>
<td>1.863</td>
<td>1.420</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The computational performance of the models reported in Table VI shows the average number of fuzzy rules/clouds/nodes, the processing time and the number of control parameters of each forecast model. The evolving models, ePFM included, have quite similar performance. ePFM and eHFN are the most compact structures for all data sets because
they achieve the smallest number of rules/nodes. Small number of rules reduces model complexity and enhances interpretability. The eHFN requires less processing time than all remaining methods for all data sets. xTS and ePFM requires the smallest and highest number of control parameters, respectively.  

Next, the forecast models are employed in management risk application using Value-at-Risk (VaR). With the VaR estimates, the models are evaluated using Violation Ratio (VR) and the Average Square Magnitude Function (ASMF). Table VII shows the values of VR and ASMF for VaR estimation using HAR, ePFM and the evolving fuzzy models considering S&P 500, Nasdaq, FTSE, DAX, IBEX and IBOVESPA stock market indexes. The evolving fuzzy models achieve better performance in terms of VR and ASMF when compared against the HAR benchmark for all economies evaluated. The ePFM presents lower values of VR and ASMF than all remaining evolving methods, except for DAX index. The ePFM reduces the VR and ASMF values in more than 48% and 65% on average when compared against HAR (Table VII), respectively. In this case, depending on the level of the portfolio exposure in monetary terms, the losses, in case of negative price movements, can be reduced significantly if ePFM for VaR modeling is adopted instead of HAR.

Summing up, the results indicate the superiority of the evolving possibilistic fuzzy approach to model and forecast realized volatility with jump components. Additionally, state of the art evolving fuzzy systems may perform as accurately as ePFM. The HAR model lacks accuracy in realized volatility predictions mainly because of its linear and static structure. The results suggest the suitability of ePFM to model distinct volatility dynamics such as in developed and emergent economies. The ePFM also has high potential for risk management such as Value-at-Risk.

VII. CONCLUSION

This paper has suggested an evolving possibilistic fuzzy modeling approach for realized volatility forecasting with jumps. The approach combines recursive possibilistic fuzzy clustering to develop the model structure, and the weighted recursive least squares to estimate the model parameters. The idea is to revise and adapt current model structure and parameters whenever required by input data. The ePFM creates new clusters and removes old ones using statistical control distance-based criteria, and identifies clusters with different shapes and orientations while processing input data recursively. The ePFM uses a utility index to evaluate the quality of the current cluster/model structure. Computational experiments with main equity market indexes of global markets show the high capability of the evolving possibilistic fuzzy approach to model realized volatility. The results show that ePFM outperforms a widely used econometric benchmark, the Heterogeneous Autoregressive model. The results also show that the ePFM surpasses several state of the art evolving fuzzy methods in VAR-based risk management task. Future work shall expand ePFM to capture mixture of cluster shapes and to make the ePFM algorithm parameter free. Additionally, the use of ePFM modeling framework in trading strategies is also a topic that deserves further investigation.

ACKNOWLEDGMENT

The authors thank the Brazilian Ministry of Education (CAPES), the Brazilian National Research Council (CNPq) grant 305906/2014-3, and the Research of Foundation of the State of São Paulo (FAPESP) for their support. They are also grateful to the anonymous reviewers for their constructive remarks which helped to improve the paper.

REFERENCES


TABLE V

<table>
<thead>
<tr>
<th>Method</th>
<th>xTS</th>
<th>eTS+</th>
<th>eCloud</th>
<th>eHFN</th>
<th>ePFM</th>
</tr>
</thead>
<tbody>
<tr>
<td>HAR</td>
<td>-7.625*</td>
<td>-7.7412</td>
<td>-5.5462*</td>
<td>-6.5263*</td>
<td>-6.5209*</td>
</tr>
<tr>
<td>xTS</td>
<td>-0.6329</td>
<td>-0.1253</td>
<td>0.9912</td>
<td>0.1982</td>
<td></td>
</tr>
<tr>
<td>eTS+</td>
<td>-0.3711</td>
<td>0.7263</td>
<td>0.9902</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eCloud</td>
<td>-0.3709</td>
<td>-0.3169</td>
<td>0.6329</td>
<td>-0.5506</td>
<td></td>
</tr>
<tr>
<td>eHFN</td>
<td>-0.0688</td>
<td>-0.3693</td>
<td>0.0991</td>
<td>0.3982</td>
<td></td>
</tr>
<tr>
<td>Nasdaq</td>
<td>-7.9834</td>
<td>0.3781</td>
<td>0.6340</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FTSE</td>
<td>-5.5273</td>
<td>-5.5273</td>
<td>-5.5273</td>
<td>-5.5273</td>
<td>-5.5273</td>
</tr>
<tr>
<td>DAX</td>
<td>-0.3727</td>
<td>-0.3727</td>
<td>-0.3727</td>
<td>-0.3727</td>
<td>-0.3727</td>
</tr>
<tr>
<td>IBEX</td>
<td>-7.3019*</td>
<td>-5.4938</td>
<td>-5.4446</td>
<td>-8.0912*</td>
<td>-5.7341*</td>
</tr>
<tr>
<td>IBOV</td>
<td>-7.4738*</td>
<td>-6.4902*</td>
<td>-6.7263*</td>
<td>-9.0013*</td>
<td>-5.4490*</td>
</tr>
</tbody>
</table>

(*) significant at 5% level.

$O(m^2 + 3m)$, where $m$ is the number of inputs.
TABLE VI

Computational complexity of the evolving fuzzy models.

<table>
<thead>
<tr>
<th>Method</th>
<th>Average # rules/clouds/nodes</th>
<th>CPU time (sec.)</th>
<th># control parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>eTS</td>
<td>10.13</td>
<td>1.0297</td>
<td>1</td>
</tr>
<tr>
<td>eF5+</td>
<td>9.74</td>
<td>0.6251</td>
<td>3</td>
</tr>
<tr>
<td>eCloud</td>
<td>8.16</td>
<td>0.3450</td>
<td>2</td>
</tr>
<tr>
<td>eHFN</td>
<td>7.35</td>
<td>0.1756</td>
<td>4</td>
</tr>
<tr>
<td>ePFM</td>
<td>6.01</td>
<td>0.2198</td>
<td>4</td>
</tr>
</tbody>
</table>

Table A

<table>
<thead>
<tr>
<th>Method</th>
<th>Average # rules/clouds/nodes</th>
<th>CPU time (sec.)</th>
<th># control parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>eTS</td>
<td>11.74</td>
<td>2.4180</td>
<td>1</td>
</tr>
<tr>
<td>eF5+</td>
<td>10.38</td>
<td>0.6372</td>
<td>3</td>
</tr>
<tr>
<td>eCloud</td>
<td>9.32</td>
<td>0.5361</td>
<td>2</td>
</tr>
<tr>
<td>eHFN</td>
<td>8.65</td>
<td>0.1934</td>
<td>2</td>
</tr>
<tr>
<td>ePFM</td>
<td>8.94</td>
<td>0.2780</td>
<td>2</td>
</tr>
</tbody>
</table>

TABLE VII

Performance evaluation of VAR estimates.

<table>
<thead>
<tr>
<th>Method</th>
<th>VAR (%)</th>
<th>ASMF (%)</th>
<th>HAR (%)</th>
<th>ASMF (%)</th>
<th>eTS (%)</th>
<th>ASMF (%)</th>
<th>eHFN (%)</th>
<th>ASMF (%)</th>
<th>ePFM (%)</th>
<th>ASMF (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>eTS</td>
<td>3.68</td>
<td>0.17</td>
<td>4.70</td>
<td>0.40</td>
<td>4.50</td>
<td>0.39</td>
<td>4.30</td>
<td>0.39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eHFN</td>
<td>3.87</td>
<td>0.17</td>
<td>4.70</td>
<td>0.40</td>
<td>4.50</td>
<td>0.39</td>
<td>4.30</td>
<td>0.39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ePFM</td>
<td>3.97</td>
<td>0.17</td>
<td>4.70</td>
<td>0.40</td>
<td>4.50</td>
<td>0.39</td>
<td>4.30</td>
<td>0.39</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Leandro Maciel received his B.Sc. degree in Economics in 2010, the M.Sc. degree in Electrical Engineering in 2012 and the Ph.D. degree in Electrical Engineering in 2015 from University of Campinas, Brazil. He is a postdoctoral research at the Institute of Economics, University of Campinas, Brazil. His interests areas include time series analysis, forecasting, finance, fuzzy systems, adaptive modeling, decision making and econometrics.

Rosangela Ballini received the B.Sc. degree in Applied Mathematics from the Federal University of So Carlos, So Carlos (USFScar), Brazil, the M.Sc. degree in Mathematics and Computation Science from the University of So Paulo (USP), So Carlos, Brazil, and the Ph. D. degree in Electrical Engineering from the School of Electrical and Computer Engineering, University of Campinas (Unicamp), Campinas, Brazil. She is Professor at the Department of Economic Theory (DTE), Institute of Economics (IE) of the University of Campinas since 2002. Her research interests include time series models, neural computation, fuzzy systems, modeling and optimization, forecasting, decision making and applications. Currently she serves on the editorial boards of IEEE Transactions on Fuzzy Systems, Applied Soft Computing and Evolving Systems. She serves the IEEE Task Forces on Adaptive Fuzzy Systems, Computational Finance and Economics Technical Committee of the IEEE Computational Intelligence Society.

Fernando Gomide received the B.Sc. degree in electrical engineering from the Polytechnic Institute of the Pontifical Catholic University of Minas Gerais (PUC/ MG) Belo Horizonte, Brazil, the M.Sc. degree in electrical engineering from the University of Campinas (Unicamp), Campinas, Brazil, and the Ph.D. degree in systems engineering from Case Western Reserve University (CWRU), Cleveland, Ohio, USA. He is professor of the Department of Computer Engineering and Automation (DCA), School of Electrical and Computer Engineering (FEEC) of the University of Campinas since 1983. His interest areas include fuzzy systems, neural, granular, and evolutionary computation, intelligent data analysis, modeling, control and optimization, decision-making and applications. He is past vice-president of IFSAS (International Fuzzy Systems Association), was IFSAS Secretary and member of the board of NAFIPS (North American Fuzzy Information Processing Society), and is former associate editor of IEEE Transactions on SMC-B and SMC-A, Fuzzy Sets and Systems, and IEEE Transactions on Fuzzy Systems. Currently he serves the editorial boards of Fuzzy Sets and Systems, Evolving Systems, Fuzzy Optimization and Decision Making, International Journal of Fuzzy Systems, Soft Computing, Granular Computation. He also is a past associate editor of Control & Automation, the journal of the Brazilian Society of Automatics (SBA), the Brazilian National Member Organization of IFAC and IFSAS. He is on the Advisory Board of the International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Journal of Advanced Computational Intelligence and Intelligent Automation and Soft Computing. He is IEEE Fellow Class 2016, member of NAFIPS, EUSFLAT, IFSAS Fellow Class 2009, and NAFIPS K. S. Fu Award 2011. He serves the IEEE Task Forces on Adaptive Fuzzy Systems, Machine Learning, and the Fuzzy Technical Council of the IEEE Computational Intelligence Society.