Risk management using evolving possibilistic fuzzy modeling

Leandro Maciel\textsuperscript{a}, Fernando Gomide\textsuperscript{a}, Rosangela Ballini\textsuperscript{b}

\textsuperscript{a}Department of Computer Engineering and Automation
School of Electrical and Computer Engineering,
University of Campinas
Av. Albert Einstein 400, Campinas, Brazil 13083–852
Emails: maciel@dca.fee.unicamp.br, gomide@dca.fee.unicamp.br

\textsuperscript{b}Department of Economic Theory
Institute of Economics
University of Campinas
R. Pitágoras 353, Campinas, Brazil 13083–857
Email: ballini@eco.unicamp.br

Abstract

Market risk exposure plays a key role for financial institutions risk management. A possible measure for this exposure is to evaluate losses likely to incur when the price of the portfolio’s assets declines using Value-at-Risk (VaR) estimates, one of the most prominent measure of financial downside market risk. This paper suggests an evolving possibilistic fuzzy modeling approach for VaR estimation. The approach is based on an extension of the possibilistic fuzzy c-means clustering and functional fuzzy rule-based modeling, which employs memberships and typicalities to update clusters and creates new clusters based on a statistical control distance-based criteria. ePFM also uses an utility measure to evaluate the quality of the current cluster structure. Computational experiments consider data of the main equity market index of Brazil, Ibovespa, from January 2000 to December 2012 for VaR estimation using ePFM, traditional econometric benchmarks such as GARCH and EWMA, and state of the art evolving approaches. The results show that ePFM is a potential candidate for VaR modeling, with better performance than alternative econometric approaches.

Keywords: Value-at-risk, fuzzy systems, risk management, evolving modeling, finance
1. Introduction

The value-at-risk, or VaR, has established itself as the most prominent measure of financial downside market risk. One of the implications of the creation of Basel Committed on Banking Supervision was implementation of VaR as the standard tool for measuring market risk, which recommended that commercial banks with significant trade activity could use their own VaR measure to define how much capital they should set aside to cover their market risk exposure, and U.S. bank regulatory agencies could audit the VaR methodology employed by the banks [35, 32].

Despite its theoretical and numerical drawbacks, such as e.g. nonsubadditive and nonconvexity [14], VaR is the most widely used risk measure in practice and its accurate computation is also fundamental for other quantile-based risk measures estimation such as expected shortfall [21]. Although VaR is a relatively simple concept, its robust estimation is often neglected in practice. Currently, the generalized autoregressive conditional heteroskedasticity (GARCH) family model [17], a type of non-linear time series model, became a standard tool for estimating the volatility of financial market data [35], which is the key input for parametric VaR computation.

It has been grown the interest in the literature of using distinct methods for VaR modeling. Examples include extreme value theory [39], quantile regression models [17], Bayesian approaches [10], and Markov switching techniques [20]. [26] summarizes an overview of these and further models for VaR estimation.

Despite the recent advances, current econometric models have some limitations. The most important is that they are based on restricted assumptions about the distribution of assets returns. There has been accumulated evidence that portfolio returns (or log returns) are usually not normally distributed. In particular, it is frequently found that market returns display structural shifts, negative skewness and excess kurtosis in the distribution of the time series [35].

To address these limitations, recent studies have suggested the use of evolving fuzzy systems (eFS) for volatility modeling and forecasting in VaR estimation [51, 40, 35, 7, 50]. More recent, [38] and [45] suggested the use of a cloud-based evolving fuzzy model and a hybrid neural fuzzy network for volatility forecasting, respectively. The authors show the high potential of evolving fuzzy modeling approaches in terms of precision, outperforming traditional econometric techniques, as well as indicate that these fuzzy
approaches appears as a potential tool to deal with volatility clustering, a financial time series stylised fact, due to their clustering-based methods for local models identification in the data space.

Evolving fuzzy systems are an advanced form of adaptive systems because they have the ability to simultaneously learn the model structure and functionality from flows of data. eFS has been useful to develop adaptive fuzzy rule-based models, neural fuzzy, and fuzzy tree models, to mention a few. Examples of the different types of evolving fuzzy rule-based and fuzzy neural modeling approaches include the pioneering evolving Takagi-Sugeno (eTS) modeling [3] approach and extensions (e.g. Simpl_eTS [4], eXtended eTS (xTS) [5]). An autonomous user-free control parameters modeling scheme called eTS+ is given in [2]. The eTS+ uses criteria such as age, utility, local density, and zone of influence to update the model structure. Later, ePL+ was developed in the realm of participatory learning clustering [36]. ePL+ extends the ePL approach [30] and uses the updating strategy of eTS+.

An alternative method for evolving TS modeling is given in [13] 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 [12]. 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 [11].

A distinct, but conceptually similar approach for TS modeling is the dynamic evolving neural fuzzy inference system model (DENFIS) [22]. 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) [34].

An on-line sequential extreme learning (OS-ELM) algorithm for single hidden layer feedforward networks with either additive or radial basis function hidden nodes in a unified framework was developed in [29]. Computational experiments using stream data of benchmark problems drawn from the regression, classification, and time series forecasting has shown that OS-ELM is an efficient tool, with good generalization and computational performance.

An evolving fuzzy modeling approach using tree structures, namely, evolv-
ing fuzzy trees (eFT) is introduced in [27]. The eFT model is a fuzzy linear regression tree whose topology can be continuously updated through a statistical model selection test. A fuzzy linear regression tree is a fuzzy tree with a linear model in each leaf. Experiments in the realm of time series forecasting have shown that the fuzzy evolving regression tree is a promising approach for adaptive system modeling.

The fuzzy self-organizing neural network [28] is an alternative recursive modeling scheme based on an error criterion, and on the generalization performance of the network. Similarly, [49] suggests a self-adaptive fuzzy inference network (SaFIN) using a categorical learning-induced partitioning mechanism to cluster data. The key is to avoid the need of prior knowledge on the number of clusters for each dimension of the input-output space. Similar mechanisms to learn model structure and parameters include the self-organizing fuzzy neural network (SOFNN) [12], and the generalized adaptive neural fuzzy inference system (GANFIS) [6]. Other important instances of evolving mechanisms include [47, 44]. A comprehensive source of the evolving approaches can be found in [33].

Recently, [37] developed a recursive possibilistic fuzzy modeling (rPFM) approach whose purpose is to improve model robustness against single-valued noise data and outliers. The ePFM uses a recursive form of the possibilistic fuzzy c-means (PFCM) clustering algorithm suggested by [41] to update the model structure, and employs the weighted recursive least squares algorithm to estimate the parameters of affine functions of the rule consequents. The PFCM simultaneously produces memberships and typicalities to alleviate outliers and noisy data sensitivity of traditional fuzzy clustering approaches, yet avoids coincident clusters [41, 25]. The advantages of PFCM have been emphasized in the literature [48, 18, 9]. Using traditional benchmarks of time series forecasting problems, [37] showed the high potential of rPFM when dealing with nonlinear and nonstationary systems affected by noise and outliers.

This paper suggests an evolving possibilistic fuzzy modeling approach (ePFM) for data streams. Despite the potential of the rPFM modeling approach to handle noisy data and outliers, it assumes that the model structure, i.e. the number of clusters or, equivalently, the number of TS fuzzy rules, is defined by the user. This is a major limitation in adaptive system modeling, especially when handling nonstationary data. To overcome this limitation, ePFM adapts its structure with a possibilistic extension of the evolving Gustafson-Kessel-Like algorithm (eGKL) suggested in [19]. Here, like [19],
creation of new clusters are determined by a statistical control distance-based criteria, but cluster structure update uses both memberships and typicalities. The model incorporates the advantages of the GK clustering algorithm of identifying clusters with different shape and orientation while processing data streams. 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 gives a simpler and more relevant rule base to encapsulate the current state of the process as mirrored by recent data.

Further, the aim of the work is also to address evolving possibilistic fuzzy modeling and value-at-risk estimation. The possibilistic approach is important because financial markets are often affected by news, expectations, and investors psychological states, which induce volatility, noisy information, and cause outliers. In this situation, possibilistic modeling has the potential to attenuate the effect of noise and outliers when building financial volatility forecasting models. Moreover, ePFM is able to handle nonlinear and time-varying dynamics such as assets returns volatility using data streams, which is essential for real-time decision making in risk management. Computational experiments were performed to compare ePFM against current econometric benchmarks and alternative state of the art evolving fuzzy and neuro-fuzzy models for VaR estimation using daily data from January 2000 to December 2012 of the main equity market index in Brazil, the Ibovespa.

After this introduction, the paper proceeds as follows. Section 2 briefly recalls TS modeling and possibilistic fuzzy c-means clustering. Next, Section 3 details the evolving possibilistic fuzzy modeling approach. Section 4 evaluates the performance and compares ePFM against GARCH, EWMA, and evolving approaches such as eTS+, ePL+, DENFIS, eFuMo, and OS-ELM. Section 5 concludes the paper and lists issues for further investigation.

2. TS model and possibilistic fuzzy c-means

This section briefly describes the basic constructs of Takagi-Sugeno modeling and its identification tasks, as well as the possibilistic fuzzy c-means clustering algorithm.
2.1. Takagi-Sugeno fuzzy model

Takagi-Sugeno (TS) fuzzy model with affine consequents consists of a set of fuzzy functional rules of the following form:

$$\mathcal{R}_i : \text{IF } \mathbf{x} \text{ is } A_i \text{ THEN } y_i = \theta_{i0} + \theta_{i1} x_1 + \ldots + \theta_{im} x_m, \quad (1)$$

where $\mathcal{R}_i$ is the $i$-th fuzzy rule, $i = 1, 2, \ldots, c$, $c$ is the number of fuzzy rules, $\mathbf{x} = [x_1, x_2, \ldots, x_m]^T \in \mathbb{R}^m$ is the input, $A_i$ is the fuzzy set of the antecedent of the $i$-th fuzzy rule and its membership function $\phi_{A_i}(\mathbf{x}) : \mathbb{R}^m \to [0, 1]$, $y_i \in \mathbb{R}$ 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.

Fuzzy inference using TS rules (1) has a closed form as follows:

$$y = \sum_{i=1}^{c} \left( \frac{\phi_{A_i}(\mathbf{x}) y_i}{\sum_{j=1}^{c} \phi_{A_j}(\mathbf{x})} \right). \quad (2)$$

The expression (2) can be rewritten using normalized degree of activation:

$$y = \sum_{i=1}^{c} \lambda_i y_i = \sum_{i=1}^{c} \lambda_i \mathbf{x}_e^T \theta_i, \quad (3)$$

where

$$\lambda_i = \frac{\phi_{A_i}(\mathbf{x})}{\sum_{j=1}^{c} \phi_{A_j}(\mathbf{x})}, \quad (4)$$

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 $\mathbf{x}_e^T = [1 \mathbf{x}^T]$ is the expanded input vector.

The TS model uses parametrized fuzzy regions and associates each region with an affine (local) model. The non-linear nature of the rule-based model emerges from the fuzzy weighted combination of the collection of the multiple local affine models. The contribution of a local model to the model output is proportional to the degree of firing of each rule.

TS modeling requires to learn the antecedent part of the model using e.g. a fuzzy clustering algorithm, and to estimate the parameters of the affine consequents.
2.2. Possibilistic fuzzy c-means clustering

The possibilistic fuzzy c-means clustering algorithm \[41\] 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 \( X = \{x_k, k = 1, \ldots, n\}, X \subseteq \mathbb{R}^{m \times n} \). The aim of clustering is to partition the data set \( X \) into \( c \) subsets (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 \( U \) (\( T \)) are the values of membership (typicalities) degrees of the data \( x \) in \( A_i \).

The possibilistic fuzzy c-means (PFCM) clustering algorithm produces \( c \) vectors that represent \( c \) cluster centers. The PFCM algorithm derives from the solution of the following optimization problem:

\[
\min_{U,T,V} \left\{ J = \sum_{k=1}^{n} \sum_{i=1}^{c} (au_{ik} + bt_{ik})D_{ik}^2 + \sum_{i=1}^{c} \sum_{k=1}^{n} \gamma_i (1 - t_{ik})^{\eta_p} \right\},
\]

subject to

\[
\begin{align*}
\sum_{i=1}^{c} u_{ik} &= 1 \forall k, \\
0 &\leq u_{ik}, t_{ik} \leq 1.
\end{align*}
\]

Here \( a > 0, b > 0, \) and \( \eta_f > 1, \eta_p > 1, \gamma_i > 0 \) are user defined parameters, and \( D_{ik}^2 \) is the distance of \( x_k \) to the \( i \)-th cluster centroid \( v_i \). The constants \( a \) and \( b \) define the relative importance of fuzzy membership and typicality values in the objective function, respectively. \( V = [v_1, v_2, \ldots, v_c]^T \in \mathbb{R}^{c \times m} \) is the matrix of cluster centers, \( \eta_f \) and \( \eta_p \) are parameters associated with membership degrees and typicalities, respectively, with default value \( \eta_f = \eta_p = 2 \).

If \( D_{ik}^2 > 0 \) for all \( i \), and \( X \) contains at least \( c \) distinct data points, then \((U,T,V) \in M_f \times M_p \times \mathbb{R}^{c \times n}\) minimizes \( J \), with \( 1 \leq i \leq c \) and \( 1 \leq k \leq n \), only if \[41\]:

\[
\begin{align*}
u_{ik} &= \left( \sum_{j=1}^{c} \left( \frac{D_{ik}}{D_{jk}} \right)^{2/(\eta_f-1)} \right)^{-1}, \\
t_{ik} &= \frac{1}{1 + \left( \frac{b}{\gamma_i} D_{ik}^2 \right)^{1/(\eta_p-1)}}.
\end{align*}
\]
\[ v_i = \frac{\sum_{k=1}^{n} (a u_{ik}^{n} + b t_{ik}^{n}) x_k}{\sum_{k=1}^{n} (a u_{ik}^{n} + b t_{ik}^{n})}, \quad (9) \]

where

\[ M_p = \left\{ T \in \mathbb{R}^{c \times n} : 0 \leq t_{ik} \leq 1, \forall i, k ; \forall k \ni i \ni t_{ik} > 0 \right\}, \quad (10) \]

\[ M_f = \left\{ U \in M_p : \sum_{i=1}^{c} u_{ik} = 1 \ \forall k ; \sum_{k=1}^{n} u_{ik} > 0 \ \forall i \right\}, \quad (11) \]

are the sets of possibilistic and fuzzy partition matrices, respectively.

Originally, [11] recommends to choose parameters \( \gamma_i \) as follows:

\[ \gamma_i = K \sum_{k=1}^{n} u_{ik}^{n} D_{ik}^2 \sum_{k=1}^{n} u_{ik}^{n}, \quad 1 \leq i \leq c, \quad (12) \]

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

3. Evolving possibilistic fuzzy modeling

The evolving possibilistic fuzzy modeling approach (ePFM) extends the evolving Gustafson-Kessel-Like clustering algorithm (eGKL), suggested in [19]. ePFM considers both membership and typicalities to update the cluster structure, i.e. the antecedents of TS fuzzy rule-based model, and incorporates a utility measure to avoid unused clusters.

3.1. Antecedents identification

ePFM proceeds based on the underlying objective function for possibilistic fuzzy clustering algorithm as in [5]. In such case, the distance \( D_{ik} \) is the same as used by the Gustafson-Kessel algorithm, i.e. the Mahalanobis distance, which is a squared inner-product distance norm that depends on a positive definite symmetric matrix \( A_{ik} \) as follows:

\[ D_{ik}^2 = \| x_k - v_i \|_{A_{ik}}^2 = (x_k - v_i)^T A_{ik} (x_k - v_i). \quad (13) \]

The matrix \( A_{ik}, i = 1, \ldots, c \), determines the shape and orientation of the cluster \( i \), i.e., it is an adaptive norm unique for every cluster, calculated by estimates of the data dispersion:

\[ A_{ik} = \left[ \rho_i \det(F_{ik}) \right]^{1/m} F_{ik}^{-1}, \quad (14) \]
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:

$$F_{ik} = \frac{\sum_{k=1}^{n} u_{ik}^{n} (x_k - v_i)(x_k - v_i)^T}{\sum_{k=1}^{n} u_{ik}^{n}}.$$  \hspace{1cm} (15)

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.

The antecedents identification of ePFM extends possibilistic fuzzy clustering algorithm using Mahalanobis distance to deal with streams of data. The evolving mechanisms, i.e. creation and update of clusters, are based on the evolving Gustafson-Kessel-Like clustering algorithm principles [19], which are inspired by two common recursive clustering algorithm: the $k$-nearest neighbor ($k$-NN) [23] and the linear vector quantization (LVQ) [24].

Suppose $x_k$ is a input data at step $k$. Two possibilities should be considered to update the current cluster structure. First, the data may belong to an existing clusters, within the cluster boundary, which requires just a clusters update. Otherwise, it may define a new cluster. These scenarios are considered in detail in what follows.

Suppose that we have $c$ clusters when the $k$-th data is input. The similarity between the new data $x_k$ and each of the existing $c$ clusters is evaluated using the Mahalanobis distance (13). The similarity relation is evaluated by checking the following condition:

$$D_{ik}^2 < \chi^2_{m,\beta}, \ i = 1, \ldots, c,$$ \hspace{1cm} (16)

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.

Relation (16) comes from statistical process control to identify variations in systems that are due to actual input changes rather than process noise. For details on criteria (16) see [19].

If the condition (16) holds, then the process is under control. The minimal distance $D_{ik}$ determines the closest cluster $p$ as;

$$p = \arg \min_{i=1,\ldots,c} (D_{ik}), \ D_{ik}^2 < \chi^2_{m,\beta}, \ i = 1, \ldots, c.$$ \hspace{1cm} (17)
In this case, the vector \( \mathbf{x}_k \) 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 \( i, i = 1, \ldots, c \).

As [19] suggests for the eGKL algorithm, the \( p \)-th cluster center could be updated using the Kohonen-like rule [24]:

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

where \( \alpha \) is a learning rate, \( v_{p,\text{new}} \) and \( v_{p,\text{old}} \) denote the new and old values of the cluster center. Notice that \( \alpha (\mathbf{x}_k - v_{p,\text{old}}) \) may be viewed as a term proportional to the gradient of an Euclidean distance based objective function, such as in FCM. The evolving possibilistic fuzzy modeling suggested in this paper updates the \( p \)-th cluster center taking into account a term proportional the gradient of the possibilistic fuzzy clustering objective function in (5). Thus, the \( p \)-th cluster center is updated as follows:

\[
v_{p,\text{new}} = v_{p,\text{old}} + \alpha (au^{q_{pk}}_{pk} + bt^{q_{pk}}_{pk}) A_{pk} (\mathbf{x}_k - v_{p,\text{old}}).
\]

The determinant and the inverse of the dispersion matrix of the \( p \)-th cluster is updated as follows:

\[
F_{p,\text{new}}^{-1} = \left( I - G_p (\mathbf{x}_k - v_{p,\text{old}}) \right) F_{p,\text{old}}^{-1} \frac{1}{1 - \alpha},
\]

\[
det(F_{p,\text{new}}) = (1 - \alpha)^{m-1} \det(F_{p,\text{old}}) (1 - \alpha + \alpha (\mathbf{x}_k - v_{p,\text{old}})^T F_{p,\text{old}}^{-1} (\mathbf{x}_k - v_{p,\text{old}})),
\]

where

\[
G_p = F_{p,\text{old}}^{-1} (\mathbf{x}_k - v_{p,\text{old}}) \frac{\alpha}{1 - \alpha + \alpha (\mathbf{x}_k - v_{p,\text{old}})^T F_{p,\text{old}}^{-1} (\mathbf{x}_k - v_{p,\text{old}})},
\]

and \( I \) is an identity matrix of order \( m \).

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

\[
v_{q,\text{new}} = v_{q,\text{old}} + \alpha (au^{q_{pk}}_{qk} + bt^{q_{pk}}_{qk}) A_{qk} (\mathbf{x}_k - v_{q,\text{old}}), \quad q = 1, \ldots, c, \quad q \neq p.
\]

\(^1\)The computation details is found in [19].
The parameter $M_i$, $i = 1, \ldots, c$, assesses the credibility of the estimated clusters. According to [19], its minimal value $M_{\text{min}}$ corresponds to the minimal number of data points needed to learn the parameters of the $i$-th inverse dispersion matrix $F_{ik}^{-1}$, that is estimated by the dimension $m$ of the data vector:

$$M_{\text{min}} = Qm(m + 1)/2,$$

(25)

where $Q$ is the credibility parameter, with default value $Q = 2$.

On the other hand, if (16) does not hold, then $x_k$ is not similar to any of the cluster centers. Thus, the natural action is to create a new cluster. However, one must check whether that fact is not due to the lack of credible clusters surrounding $x_k$, that is, whether the condition:

$$M_p < M_{\text{min}},$$

(26)

applies, $p = \arg\min_{i=1,\ldots,c} (D_{ik})$. If this is the case, then the closest cluster $p$ is updated using (20)-(23). Otherwise, a new cluster is created, $c_{\text{new}} = c_{\text{old}} + 1$, with the following initialization:

$$v_{c,\text{new}} = x_k, \quad F_{c,\text{new}}^{-1} = F_0^{-1} = \kappa I, \quad \det(F_{c,\text{new}}) = \det(F_0), \quad M_{c,\text{new}} = 1,$$

(27)

where $I$ is an identity matrix of size $m$ and $\kappa$ is a sufficient large positive number.

The initialization in (27) is also used if there is no initially collected data set, supposing that $x_1$ represents the very first data point of the data streams $x_k$, $k = 1, 2, \ldots$.

3.2. Cluster quality measurement

In this paper, the quality of the cluster structure is monitored at each step considering the utility measure introduced in [2]. The utility measure is an indicator of the accumulated relative firing level of a corresponding rule:

$$U_{ik} = \frac{\sum_{i=1}^{k} \lambda_i}{k - I^*},$$

(28)

where $I^*$ is the step that indicates when cluster $i*$ was created.

Once a rule is created, the utility indicates how much the rule has been used. This quality measure aims at avoiding unused clusters kept in the structure. Clusters corresponding with low quality fuzzy rules can be deleted. Originally, [2] suggested, at each step $k$, the following criteria: if $U_{ik}$ is less or
equal than a threshold, specified by the user, then the \( i \)-th cluster is removed.

To turn the ePFM algorithm more autonomous, the following criteria to remove low quality clusters is suggested:

\[
\text{If } U_{ik} \leq (\bar{U}_i - 2\sigma_{U_i}) \text{ Then } c_{\text{new}} = c_{\text{old}} - 1,
\]

(29)

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

This criteria means that if the utility of cluster \( i \) at \( k \) is less or equal than 2 standard deviation of the average utility of cluster \( i \), then cluster \( i \) has low utility and it is removed. This idea relates to the 2\( \sigma \) process control band in statistical process control but considering the tail of the left side of utility distribution, since it is associated with the most unused clusters (lower utility). This principle guarantees high relevance cluster structure and corresponding fuzzy local models. Alternative quality measures such as age, support, zone of influence and local density may be adopted.

3.3. Consequent parameters estimation

Estimation of the parameters of the affine rule consequents is done using weighted recursive least squares algorithm (wRLS) \[31\] as in \[2\]. Expression \[3\] can be rewritten as:

\[
y = \Lambda^T \Theta,
\]

(30)

where \( \Lambda^T = [\lambda_1 x_e^T, \lambda_2 x_e^T, \ldots, \lambda_c x_e^T] \) is the fuzzily weighted extended input, \( x_e^T = [1 \ x^T] \) the expanded data vector, and \( \Theta^T = [\theta_1^T, \theta_2^T, \ldots, \theta_c^T] \) the parameter matrix, \( \theta_i^T = [\theta_0, \theta_1, \ldots, \theta_m] \).

Given that the actual output can be obtained at each step, the parameters of the consequents can be updated using the recursive least squares (RLS) algorithm considering local or global optimization. In this paper we use the locally optimal error criterion wRLS:

\[
\min_{\theta_i} E^i_L = \min_{\theta_i} \sum_{k=1}^{n} \lambda_i \left( y_k - x_{ek}^T \theta_{ik} \right)^2.
\]

(31)

Therefore, parameters of the rule consequents are updated as follows \[31, 2\]:

\[
\theta_{i,k+1} = \theta_{i,k} + \Sigma_{ik} x_{ek} \lambda_i \left( y_k - x_{ek}^T \theta_{ik} \right), \quad \theta_{0,0} = 0,
\]

(32)

\[
\Sigma_{i,k+1} = \Sigma_{i,k} - \frac{\lambda_{ik} \Sigma_{ik} x_{ek} x_{ek}^T \Sigma_{ik}}{1 + \lambda_{ik} x_{ek} x_{ek}^T \Sigma_{ik}}, \quad \Sigma_{0,0} = \Omega I,
\]

(33)
where $I$ is an identity matrix of size $m$, $\Omega$ a large number (usually $\Omega = 1000$), and $\Sigma$ the dispersion matrix.

### 3.4. ePFM algorithm

The evolving possibilistic fuzzy modeling (ePFM) approach is summarized next. The steps of the algorithm are non-iterative. The procedure adapts an existing model whenever the pattern encoded in data changes. Its recursive nature means that, as far as data storage is concerned, it is memory efficient.

**Evolving possibilistic fuzzy modeling**

1. Compute $M_{\text{min}} = Qm(m+1)/2$.
2. Choose control parameters $a$, $b$, $\alpha$, $\beta$, $\kappa$, and initialize $F_{-1}^{-1} = \kappa I$.
3. **For** $k = 1, 2, \ldots$ **do**
4. read the next data $x_k$
5. check the similarity of $x_k$ to existing clusters: $D_{ik}^2 < \chi^2_{m,\beta}$
6. identify the closest cluster: $p = \arg \min_{i=1,\ldots,c} (D_{ik})$
7. **if** $D_{ik}^2 < \chi^2_{m,\beta}$ or $M_p < M_{\text{min}}$ **then**:
   a. update the parameters of the $p$-th cluster using (20)-(23)
   b. move away the centres of remaining cluster using (24)
8. **go to step** 4
9. **else**
10. create a new cluster: $c_{\text{new}} = c_{\text{old}} + 1$
11. initialize the new cluster using (27)
12. **end if**
13. **if** $U_{ik} \leq (\bar{U}_k - 2\sigma_{U_k})$ **then** delete cluster $i$:
14. $c_{\text{new}} = c_{\text{old}} - 1$
15. **end if**
16. compute rule consequent parameters using the wRLS
17. compute model output $y_{k+1}$
18. **end for**

### 4. Computational experiments

The ePFM approach introduced in this paper gives a flexible modeling procedure and can be applied to a range of problems such as process modeling, time series forecasting, classification, system control, and novelty detection. This section evaluates the performance of ePFM for Value-at-Risk
modeling and estimation of Ibovespa index in terms of volatility forecasting. The results of ePFM is compared with GARCH and EWMA models, and with state of the art evolving fuzzy, neuro and neuro-fuzzy modeling approaches, the eTS+ [2], ePL+ [36], rFCM [13], DENVIS [22], eFuMo [11], and OLS-ELM [29].

4.1. VaR 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.:

\[
\Pr(r_{k+h} \leq \text{VaR}_{k+h}^{\alpha_{VaR}}) = 1 - \alpha_{VaR}
\]

where

\[
r_{k+h} = \frac{\ln(P_{k+h})}{\ln(P_k)},
\]

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

Let us assume that the daily conditional heteroskedastic returns in (35) of a financial asset can be described by the following process:

\[
r_k = \sigma_k z_k,
\]

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

Therefore, the VaR at \( k + 1 \) is given by:

\[
\text{VaR}_{k+1}^{\alpha_{VaR}} = \sigma_{k+1}CDF_z^{-1}(\alpha_{VaR}),
\]

where \( CDF_z^{-1}(\alpha_{VaR}) \) is the critical value from the normal distribution table at \( \alpha_{VaR} \) confidence level. In this section \( \alpha_{VaR} = 5\% \) confidence level is assumed for all models.

In a VaR forecasting context, volatility modeling plays a crucial role and thus it should place emphasis on the volatility models implemented. This work chooses to model the conditional variance of the returns process with two econometric volatility models and some evolving methods.
4.2. Econometric benchmarks

Two benchmark econometric approaches, GARCH and EWMA, are adopted to estimate the VaR given a $\alpha_{VaR}$ confidence level in (37). The GARCH($r$, $s$) model is as follows [16]:

\[
\sigma_k^2 = \delta_0 + \sum_{j=1}^{r} \delta_j r_{k-j}^2 + \sum_{l=1}^{s} \mu_l \sigma_{k-l}^2,
\]

where $\delta_j$, $j = 0, 1, \ldots, r$, and $\mu_l$, $l = 1, 2, \ldots, s$, are model parameters.

The exponentially weighted moving average (EWMA) model of Riskmetrics has the following form [43]:

\[
\sigma_{k+1}^2 = \lambda^E \sigma_k^2 + (1 - \lambda^E) r_k^2,
\]

where $\lambda^E$ is the forgetting factor with default value $\lambda^E = 0.94$.

The evolving models use the following representation to estimate VaR:

\[
\sigma_{k+1}^2 \approx r_{k+1}^2 = f(r_k^2, r_{k-1}^2, \ldots, r_{k-p}^2),
\]

where $p$ is the number of lags considered as input of the evolving models. The number of lags is chosen looking at the partial autocorrelation function of the squared indexes returns.

4.3. Performance evaluation

The performance of the models is evaluated 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 greater than the estimated maximum loss in the VaR framework. VR is computed as follows:

\[
VR = \frac{1}{n} \sum_{k=1}^{n} \Phi_k,
\]

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 $n$ is the number of observations in the test set.

The average square magnitude function (ASMF) [15] considers the amount of possible default measuring the average squared cost of exceptions. It is computed using:

\[
\text{ASMF} = \frac{1}{\vartheta} \sum_{j=1}^{\vartheta} \xi_j,
\]
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 [35].

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

### 4.4. Data

The computational results were produced using daily values of the Ibovespa index from January 2000 to December 2012[2]. This period was selected because the Brazilian equity market was under stable movements and also volatile dynamics due to the recent international crisis. The Ibovespa index illustrates how the models perform in emergent economies like the Brazilian. The data was split in two sets. The training set includes the period from January 2000 to December 2003. The remaining data comprises the test set.

### 4.5. Results

Summary statistics of Ibovespa index returns are presented in Table [1]. For the index returns serie the mean is close to zero and also presented a negative skewness, which indicates left a side fat tail, a stylized fact of financial assets returns. The serie shows high kurtosis coefficient as well. The Jarque-Bera statistic (JB) [8] reveals that the return serie is non-normal with a 99% confidence interval. Figure [1] shows the returns serie of Ibovespa. In this market volatility clusters are presented, mainly in cases of more unstable periods, such as during the recent crisis that started in 2008 is USA and then affected other economies like the Brazilian.

---

[2]The data was provided by Bloomberg.
Table 1: Statistics of Ibovespa index returns.

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Ibovespa</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.0004</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>0.0191</td>
</tr>
<tr>
<td>Skewness</td>
<td>-0.1177</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>6.7847</td>
</tr>
<tr>
<td>Max</td>
<td>0.1368</td>
</tr>
<tr>
<td>Min</td>
<td>-0.1210</td>
</tr>
<tr>
<td>JB</td>
<td>1928.0</td>
</tr>
<tr>
<td>p-value</td>
<td>0.0010</td>
</tr>
</tbody>
</table>

The parameters \( r \) and \( s \) of the GARCH\((r,s)\) model were selected according to the Bayesian Information Criteria (BIC) \[46\]. The final GARCH model, GARCH\((1,1)\) for Ibovespa index returns is:

\[
\sigma_k^2 = .5029r_{k-1}^2 + .4230\sigma_{k-1}^2.
\] (43)

The GARCH model indicates that Ibovespa returns volatility is very persistent due to its estimated coefficients \((\delta_1 + \mu_1 = .5029 + .4230 = .9592)\), which is in line with the literature for volatility behavior in emergent economies \[1\].

For the evolving methods, the analysis of the partial autocorrelation function of the squared index returns indicated \( p = 4 \) for Ibovespa VaR estimation:

\[
\sigma_{k+1}^2 \approx r_{k+1}^2 = f(r_k^2, r_{k-1}^2, r_{k-2}^2, r_{k-3}^2, r_{k-4}^2).
\] (44)
The control parameters of ePFM were chosen based on experiments conducted to find the best performance in terms of VR and ASMF measures. ePFM modeling uses $a = 1$, $b = 2$, $\alpha = 0.13$ and $\kappa = 50$. One must note that is necessary to choose the probability of a false alarm $\beta$ and to define $\chi^2_{m,\beta}$. 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 [19]. Table 2 show the $\chi^2_{m,0.0455}$ for different values of $m$. Therefore, ePFM uses $\chi^2_{5,0.0455} = 11.3139$. Control parameters of the alternative evolving methods were also chosen based on simulations to find the best VR and ASMF values.

\begin{table}[h]
\centering
\caption{$\chi^2_{m,0.0455}$ for different values of $m$.}
\begin{tabular}{cccccccc}
\hline
$m$ & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
\hline
$\chi^2_{m,0.0455}$ & 6.1801 & 8.0249 & 9.7156 & 11.3139 & 12.8489 & 14.3371 & 15.7891 \\
\hline
\end{tabular}
\end{table}

Tables 3 shows the violation ratio VR and average square magnitude function ASMF values of the ePFM against the remaining approaches for Ibovespa VaR estimation using test data. The ePFM model achieves competitive results in terms of VR and ASMF when compared against eFuMo and remaining evolving models (Table 3). The GARCH and EWMA models achieve the worst performance, with higher values of violation ratio and average squared magnitude function. The evolving possibilistic fuzzy approach reduces the VR and ASMF values in approximately 62.17% and 72.82%, respectively, when compared against the econometric benchmark models, GARCH and EWMA (Table 3). These results are in line with [38] and [45] which suggest that clustering based techniques perform better for volatility modeling and forecasting by handling volatility clustering, since in this empirical study eTS+, ePL+, rFCM, eFuMo and ePFM show the lowest values of VR and ASMF against the other models.
Table 3: Performance evaluation for Ibovespa index VaR estimation.

<table>
<thead>
<tr>
<th>Method</th>
<th>VR(%)</th>
<th>ASMF(%)</th>
<th># rules (aver.)</th>
<th>Time (sec.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GARCH</td>
<td>4.367</td>
<td>0.214</td>
<td>-</td>
<td>46.34</td>
</tr>
<tr>
<td>EWMA</td>
<td>4.567</td>
<td>0.279</td>
<td>-</td>
<td>7.019</td>
</tr>
<tr>
<td>eTS+</td>
<td>1.982</td>
<td>0.106</td>
<td>4.871</td>
<td>1.876</td>
</tr>
<tr>
<td>ePL+</td>
<td>2.014</td>
<td>0.117</td>
<td>6.740</td>
<td>1.994</td>
</tr>
<tr>
<td>rFCM</td>
<td>2.276</td>
<td>0.098</td>
<td>5.306</td>
<td>2.035</td>
</tr>
<tr>
<td>DENFIS</td>
<td>3.009</td>
<td>0.192</td>
<td>15</td>
<td>21.48</td>
</tr>
<tr>
<td>eFuMo</td>
<td>1.874</td>
<td>0.079</td>
<td>4.652</td>
<td>1.715</td>
</tr>
<tr>
<td>OS-ELM</td>
<td>2.933</td>
<td>0.164</td>
<td>13</td>
<td>16.30</td>
</tr>
<tr>
<td>ePFM</td>
<td>1.667</td>
<td>0.085</td>
<td>4.891</td>
<td>1.894</td>
</tr>
</tbody>
</table>

Except for GARCH, EWMA, DENFIS and OS-ELM, the computational performance of the evolving approaches are similar in terms of CPU time, considering all test data processing (Table 3). eFuMo develops the smallest average number of rules among all approaches. A smaller number of rules reduces model complexity and enhances interpretability. Generally speaking, all evolving approaches are qualified to deal with on-line stream data processing for VaR estimation in risk management decision making.

Figure 2 shows the returns and VaR estimates produced by ePFM modeling approach for Ibovespa index. Notice the high adequacy of ePFM to capture volatility dynamics in terms of VaR estimates.

Figure 2: Daily Ibovespa returns and VaR estimates using ePFM.
5. Conclusion

This paper has suggested an evolving possibilistic fuzzy modeling approach and its application in Value-at-Risk (VaR) estimation. The approach combines recursive possibilistic fuzzy clustering to learn the model structure, and a weighted recursive least squares to estimate the model parameters. The idea is to adapt the model structure and parameter whenever required by new input data. ePFM creates new clusters using a statistical control distance-based criteria, and clusters are updated using memberships and typicalities. The model incorporates the advantages of the Gustafson-Kessel clustering algorithm of identifying clusters with different shape and orientation while processing data streams. ePFM also uses an utility measure to evaluate the quality of the current cluster/model structure. Computational experiments addressed ePFM, econometric GARCH and EWMA benchmarks, and state of the art evolving fuzzy and neuro-fuzzy models for VaR estimation using daily data from January 2000 to December 2012 of the main equity market indexes in Brazil (Ibovespa). Results indicate the superior performance of ePFM and all the evolving techniques against GARCH and EWMA benchmarks in both economies considered. Further work shall generalize ePFM to handle mixture of cluster shapes, to make the ePFM algorithm fully autonomous, and to evaluate the ePFM model in risk management strategies.

References


