Expert Systems with Applications 39 (2012) 12302–12309

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Financial time series forecasting with a bio-inspired fuzzy model

José Luis Aznarte [a,*], Jesús Alcalá-Fdez [b], Antonio Arauzo-Azofra [c], José Manuel Benítez [b]

[a] Center for Energy and Processes (CEP), Mines ParisTECH, Sophia Antipolis, France
[b] Department of Computer Science and Artificial Intelligence, University of Granada, Spain
[c] Department of Civil Engineering, University of Córdoba, Spain

ARTICLE INFO

Keywords:
Time series forecasting
Fuzzy rule-based systems
Regime switching models
Financial time series

ABSTRACT

In general, times series forecasting is considered as a highly complex problem, which is particularly true for financial time series. In this paper, a fuzzy model evolved through a bio-inspired algorithm is proposed to produce accurate models for the prediction of these time series. The performance of this model is compared to that of a group of state-of-the-art statistical models. A thorough experimental study is designed and carry out in order to assess the merits of the proposal. The experimental results allow us to state that our proposal forecasts consistently outperform the other considered methods.

1. Introduction

In general, time series forecasting is considered as a highly complex problem. This is true in particular when dealing with financial time series, as pointed out by Pissarenko (2002).

The predictability of some common financial time series, as stock prices or level of indices, is a controversial issue which has been questioned in the past. These series tend to behave as random-walk processes which, in theory, makes its prediction impossible. On the other hand, financial time series are usually subject to regime shifting, which means that their statistical properties do not remain constant through time. For example, some series do react to business cycle or to slow-down periods.

Furthermore, a large amount of random day-to-day variations are usually found in these series, which makes them still more unpredictable. Examples of events which influence the behaviour of a financial series are the announcements of firm specific news, fiscal measures, employment reports as well as political events. The effect of this on a specific series is to increase its noisy nature, making it difficult to distinguish good prediction algorithms from bad ones, since even a naïve predictor can produce good results. See Hellstrom and Holmstrom (1998) for a detailed discussion.

The modeling and forecasting of time series is a focus problem in a number of areas, traditionally Statistics and – in derived – Econometrics. The Soft Computing field, however, has also shown interest in the problem, and the papers paying attention to it are increasingly growing in number. The problem has been approached with different kind of proposals: neural networks, fuzzy systems, and hybridizations. Some examples, Zhang (2007), Rivas, Merelo, Castillo, Arenas, and Castellano (2004), da Silva (2008), Chen, Yuan, Dong, and Abraham (2005). But also, evolutionary algorithms have been considered, just like those proposed in Tang and Xu (2005), Armano, Marchesi, and Murru (2005), and Sheta and Jong (2001). The proposal we bring in this work is of this last kind. The method we propose includes all the steps for the complete development of a forecaster: since the model selection (building) to the estimation (tuning and simplification). The complete method is engineered towards achieving the highest accuracy.

On the other hand, the traditional area concerned with the study of time series is Statistics. Along the time a rather lengthy list of models have been proposed. The initial proposal were of a linear nature, the most salient of which is the well-known ARMA model by Box and Jenkins (1970). However, due to the its linear character it does pose a severe limitation when dealing with nonlinear time series. To address those cases a number of models have been proposed. In particular regime switching models have receive a great deal of attention. In this paper we consider the most salient of those.

This paper is structured as follows. The method that we proposed is presented and discussed in Section 2. Statistical methods for time series forecasting are detailed in Section 3. An extensive empirical evaluation is described and developed in Section 4. The paper finishes with the conclusions drawn upon the research being reported.

2. A bio-inspired fuzzy model for financial time series

Fuzzy sets were proposed by Zadeh (1965) as a means to represent and infer with knowledge affected by vagueness. Fuzzy systems have arisen as an effective tool for modeling, especially, for complex systems. We consider a fuzzy rule-based system as a base model for time series forecasting. A key issue when using fuzzy

* Corresponding author.
E-mail address: jose-luis.aznarte@mines-paristech.fr (J.L. Aznarte).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.02.135

systems is how to properly build them. For that we propose a bio-inspired algorithm. The whole compound is called LEL-TSK.

2.1. Fuzzy rule-based model

Takagi and Sugeno (1985) and Sugeno and Kang (1988), presented a mathematical tool to procure a fuzzy model of a system. They suggested a multidimensional fuzzy reasoning in which the number of implications can be surprisingly decreased; that is, we would need fewer rules in the knowledge base.

The fuzzy model is based on rules in which the consequent is not a linguistic variable (as in Mamdani fuzzy systems) but a function of the input variables. In the time series framework, these kind of rules usually presents the following structure:

$$\text{IF } y_{t-1} \text{ IS } A_1 \text{ AND } y_{t-2} \text{ IS } A_2 \text{ AND } \dots \text{ AND } y_{t-p} \text{ IS } A_p$$
$$\text{THEN } \hat{y}_i = \mathbf{b}\mathbf{x}_t = b_0 + b_1 y_{t-1} + b_2 y_{t-2} + \cdots + b_p y_{t-p}, \quad (1)$$

where the vector $\mathbf{x}_t$ is composed of the variables y_{t-i}, which are lagged values of the time series, $\{y_t\}$.

The output of an FRBM considering a knowledge base composed of m TSK rules is computed as the weighted average of the individual rule outputs $\hat{y}_i$, $i = 1 \dots m$:

$$y = \frac{\sum_{i=1}^{r} \mu_i \mathbf{b}_i \mathbf{x}_t}{\sum_{i=1}^{r} \mu_i}, \quad (2)$$

where μ_i is the firing strength of each rule and is calculated as the T-norm (product) of the membership degree μ_{A_i} of each lagged variable y_{t-i} to a fuzzy term A_i:

$$\mu_i = \prod \left(\mu_{A_1}(y_{t-1}), \mu_{A_2}(y_{t-2}), \dots, \mu_{A_p}(y_{t-p}) \right). \quad (3)$$

2.2. An evolutionary algorithm for FRBM building and estimation

In Alcalá, Alcalá-Fdez, Casillas, Cordón, and Herrera (2007), the use of local semantics-based Mamdani fuzzy rules as local fuzzy prototypes was proposed to obtain accurate local semantics-based TSK rules, considering the interaction between input and output variables and taking into account the fuzzy nature of these kind of rules. To do so, a two-stage genetic FRBM was designed following the MOGUL paradigm (Cordón, del Jesus, Herrera, & Lozano, 1999), a methodology to obtain genetic FRBMs.

The local identification of prototypes induces competition amongst rules, considering only the quality of the approximation performed by each rule. However, the global cooperation among rules should be considered in order to increase the generalization ability of the model. Following the MOGUL approach, a post-processing stage is considered for this purpose. In this way, the learning method substantially reduces the search space size by dividing the genetic learning process into two stages. A complete description of this algorithm can be found in Alcalá et al. (2007). A brief description of this technique is presented below.

2.2.1. Local process for identifying prototypes

To obtain a set of local semantics-based Mamdani fuzzy rules (fuzzy prototypes) a method described in Cordón and Herrera (2001) that is based on local covering measures to induce competition among rules can be used, considering the *completeness* and *consistency* properties (Cordón & Herrera, 1999). In this case, completeness is verified requiring that each example is covered to a degree $\epsilon \in \mathbb{R}^+$. On the other hand, to verify the consistency, the *positive* and *negative example* concepts (Cordón & Herrera, 1999) are considered. Thereby, the accuracy of a simple fuzzy rule, R_h, on the set of examples, E, is measured by using a multicriteria fitness function:

$$F(R_h) = \Psi_E(R_h) \cdot G_\omega(R_h) \cdot g_n(R_h^-), \quad (4)$$

designed to take into account three different criteria (Cordón & Herrera, 1999): *high frequency value* ($\Psi_E(R_h)$), *high average covering degree over positive examples* ($G_\omega(R_h)$) and *small negative example set* ($g_n(R_h^-)$). This process is briefly summarized in the following steps:

1. Perform a grid-style (global semantics) fuzzy partition for each variable.
2. Generate for each example e_l the global semantics-based fuzzy rule best covering it. Then, evaluate all the global fuzzy rules and select the rule with the highest value in the fitness function ($F(R_h)$).
3. The most promising global fuzzy rule is locally tuned to identify the local fuzzy prototype best grouping the data located in the corresponding subspace. This process is computed by means of the (1+1)-Evolutionary Strategy ((1+1)-ES) described in Cordón and Herrera (2001) considering as fitness function $F'(R_h) = F(R_h) \cdot LNIR(R_h)$ where $LNIR(R_h)$ is a penalty function to avoid excessive proximity among prototypes (Cordón & Herrera, 1999).
4. Finally, the obtained prototype is added to the final set of fuzzy prototypes. Data covered to a certain degree by this set are removed and not considered for future iterations. The iterative process ends up when no more uncovered training data remains.

To obtain the TSK consequents, once the set of local fuzzy prototypes is obtained and considering the same antecedents, the existing partial linear input–output relation is computed using the data located in each input subspace by means of the (μ, λ)-ES presented in Cordón and Herrera (1999) to minimize the Mean Square Error (MSE).

2.2.2. Post-processing stage

Two different processes are considered at this stage to improve the MSE: the *genetic simplification process* and the *genetic tuning process*:

1. *Genetic simplification process*: This process, described in Cordón and Herrera (2001), is based on a standard binary-coded GA and also considers the completeness property. It has the aim of selecting the subset of rules cooperating best among the rules generated in the previous stage.
2. *Genetic tuning process*: This method is an adaptation of Cordón and Herrera (1999) (tunes TSK FRBM based on global semantics) to local semantics. It is based on a hybrid GA-ES algorithm in which each individual represents a complete knowledge base. An (1+1)-ES is considered as a genetic operator to locally tune a percentage δ of the best individuals in each generation. In this method, the variation interval is independently estimated for each fuzzy set.

3. Regime switching models

In statistical time series modeling, one of the oldest and most successful concepts is to forecast future values of a time series as a combination of its past values. This is a quite natural idea that we apply on every day's life, and it was popularized in 1970 after (Box & Jenkins, 1970). In that work, Box and Jenkins formalized the use of the *autoregressive moving average* (ARMA) model, and its special case the *autoregressive* (AR) model which assumes that future values of a time series can be expressed as a linear combination of its past values.

An AR model of order $p \geqslant 1$ is defined as

$$X_t = b_1 X_{t-1} + \cdots + b_p X_{t-p} + \varepsilon_t, \quad (5)$$

where $\{\varepsilon_t\} \sim N(0,\sigma^2)$. For this model we write $\{X_t\} \sim$ AR (p), and the time series $\{X_t\}$ generated from this model is called the AR (p) process.

Such a simple model proved to be extremely useful and suited to some series which, at first sight, seemed to be too complex as to be linear. Applications of the Box and Jenkins methodology spread in the following decades, covering various scientific areas such as Biology, Astronomy or Econometrics.

However, there were still many problems which could not be modeled using linear models. In 1978, taking a step towards non-linearity, Tong (1978) proposed a *piecewise linear* model: the threshold autoregressive (TAR) model, which is based on the idea of partitioning the state-space into several subspaces, each of which was to be modeled by an AR model. To control the transitions from one linear model to another, a set of thresholds must be defined on one of the variables involved. This variable can be an exogenous variable associated to the process being modeled or one of the lagged values of the series, in which case the model is called self-exciting—yielding the acronym SETAR.

A *threshold autoregressive* (TAR) model with k ($k \geqslant 2$) regimes is defined as

$$y_t = \sum_{i=1}^{k} \boldsymbol{\omega}_i \mathbf{x}_t I(s_t \in A_i) + \varepsilon_t = \sum_{i=1}^{k} \{\omega_{i,0} + \omega_{i,1} y_{t-1} + \cdots + \omega_{i,p_i} y_{t-p_i}\} I(s_t \in A_i) + \varepsilon_t, \quad (6)$$

where s_t is the threshold variable, I is an indicator (or *step*) function, $p_1, \ldots, p_k$ are some unknown positive integers, $\boldsymbol{\omega}_i$ are unknown parameters, and $\{A_i\}$ forms a partition of $(-\infty,\infty)$ with $\cup_{i=1}^{k} A_i = (-\infty,\infty)$ and $A_i \cap A_j = \emptyset$, $\forall i \neq j$.

In this model, we fit on each subset A_i a linear autoregressive form. The partition is dictated by the threshold variable s_t which can be a lagged value of the series (y_{t-d}). Usually, $A_i = r_{i-1}, r_i]$, with $-\infty = r_0 < r_1 < \cdots < r_k = \infty$, where the r_i's are called thresholds. Fig. 1 shows an example of a one-dimensional TAR model with 2 regimes.

3.1. Smooth transition autoregressive model (STAR)

A key feature of TAR models is the discontinuous nature of the AR relationship as the threshold is passed. Taking into account that nature is generally continuous, in 1994 an alternative model called *smooth transition autoregressive* (STAR) was proposed by Teräsvirta (1994). In STAR models there is a smooth continuous transition from one linear AR to another, rather than a sudden jump.

In this model and variants (cf. van Dijk, Teräsvirta, & Franses, 2002), the indicator function $I(\cdot)$ in (6), which, as shown above, is a *step* function that takes the value zero below the threshold and one above it is substituted by a smooth function with sigmoid characteristics. The STAR model with k regimes ($k > 2$) is defined as

$$y_t = \sum_{i=1}^{k} \boldsymbol{\omega}_i \mathbf{x}_t \mathrm{F}_i(s_t; \boldsymbol{\phi}_i) + \varepsilon_t. \quad (7)$$

The transition function, F ($s_t; \boldsymbol{\phi}_i$), is a continuous function that is bounded between 0 and 1. The regime that occurs at time t is determined by the observable variable s_t and the associated value of F($s_t; \boldsymbol{\phi}_i$). Different choices for the transition function give rise to different types of regime-switching behavior. A popular choice for F($s_t; \boldsymbol{\phi}_i$) is the first-order logistic function,

$$f(y_{t-d}; c, \gamma) = (1 + \exp(-\gamma(y_{t-d} - c)))^{-1} \quad (8)$$

and the resultant model is called the logistic STAR (LSTAR) and an example of a two regime LSTAR is shown in Fig. 2.

In the LSTAR model, we define the transition function F($s_t; \boldsymbol{\phi}_i$) of expression (7) as

$$\mathrm{F}_i(s_t; \boldsymbol{\phi}_i) = \begin{cases} 1 - f(s_t; \gamma_i, c_i) & \text{if } i = 1, \\ f(s_t; \gamma_i, c_i) - f(s_t; \gamma_{i+1}, c_{i+1}) & \text{if } 1 < i < k, \\ f(s_t; \gamma_i, c_i) & \text{if } i = k, \end{cases} \quad (9)$$

where $f(s_t;\gamma_i,c_i)$ is defined as in (8). The LSTAR model can be (and usually is) consequently rewritten as

$$y_t = \boldsymbol{\omega}_1 \mathbf{x}_t + \sum_{i=2}^{k} \boldsymbol{\omega}_i \mathbf{x}_t f(s_t; \gamma_i, c_i) + \varepsilon_t. \quad (10)$$

Each of the parameters c_i in (10) can be interpreted as the threshold between two regimes, in the sense that the logistic function changes monotonically from 0 to 1 as s_t increases and F ($c_i; \gamma_i, c_i$) = 0.5. The parameter γ_i determines the smoothness of the transition from one regime to another. As γ_i becomes very large, the logistic function approaches the indicator function $I(\cdot)$ and hence the change of F ($s_t; \gamma_i, c_i$) from 0 to 1 becomes instantaneous at $s_t = c$. Consequently, the LSTAR nests threshold autoregressive (TAR) models as a special case. Furthermore, when $\gamma \to 0$ the LSTAR model reduces to a linear AR model.

In the LSTAR model, the regime switches are associated with small and large values of the transition variable s_t relative to c. In certain applications it may be more appropriate to specify the transition function such that the regimes are associated with small and large absolute values of s_t (again relative to c). This can be achieved by using, for example, the exponential function, in which case the model may be named ESTAR. Other frequently used function is the normal distribution, which yields the acronym NSTAR.

3.2. Neuro-Coefficient smooth transition autoregression (NCSTAR)

One of the latest developments in threshold-based models is the Neuro-Coefficient STAR (Medeiros & Veiga, 2005). This model is a generalization of the previously described models and can handle multiple regimes and multiple transition variables. It can be seen as a linear model whose parameters change through time and are determined dynamically by a single hidden layer feed-forward neural network.

Consider a linear model with time-varying coefficients expressed as

$$y_t = \boldsymbol{\phi}_t' \mathbf{x}_t + \varepsilon_t, \quad (11)$$

where $\boldsymbol{\phi}_t = (\phi_t^{(0)}, \phi_t^{(1)}, \ldots, \phi_t^{(p)})' \in \mathbb{R}^{p+1}$ is a vector containing the coefficients of the model. The time evolution of the coefficients $\phi_t^{(j)}$ of (11) is given by the output of a single hidden layer neural network with k hidden units

$$\phi_t^{(j)} = \sum_{i=1}^{k} v_{ji} f(\boldsymbol{\omega}_i \mathbf{z}_t) - v_{j0}, \quad j = 0, \ldots, p, \quad (12)$$

where v_{ji} and v_{j0} are real coefficients.

Substituting the p realizations of (12) in (11) we obtain the general form of the NCSTAR model:

$$y_t = \boldsymbol{v}_1 \mathbf{x}_t + \sum_{i=2}^{k} \boldsymbol{v}_i \mathbf{x}_t f(\boldsymbol{\omega}_i \mathbf{z}_t) + \varepsilon_t, \quad (13)$$

where $\mathbf{z}_t$ is a $q \times 1$ vector of transition variables and $\omega_i = [\omega_{1i}, \ldots, \omega_{qi}]'$ are real parameters. The norm of ω_i, called γ_i, is known as the *slope parameter*. In the limit, when the slope parameter approaches infinity, the logistic function becomes a step function. The function f is defined as in (8), but if the alternative smoothing function F, as defined in (9), is used, the model can be rewritten as

$$y_t = \sum_{i=1}^{k} \boldsymbol{v}_i \mathbf{x}_t \mathrm{F}_i(\boldsymbol{\omega}_i \mathbf{z}_t) + \varepsilon_t. \quad (14)$$

The choice of the elements of $\mathbf{z}_t$, which determines the dynamics of the process allows a number of special cases. An important one is when $\mathbf{z}_t = y_{t-d}$. In this case, model (13) becomes a LSTAR model with k regimes, expressed as in (10). It should be noticed as well that this model also nests the SETAR model. When $\gamma_i \to \infty \, \forall i$, the LSTAR model becomes a SETAR model with k regimes.

Another interesting case is when $\boldsymbol{v}_i' = (v_{0i}, 0, \ldots, 0)$. Then the model becomes an AR-NN model (Suarez-Farinas, Pedreira, & Medeiros, 2004) with k hidden units. Finally, this model is related to the Functional Coefficient Autoregressive (FAR) model (Chen & Tsay, 1993), and to the Single-Index Coefficient Regression model (Xia & Li, 1999).

4. Empirical evaluation

In order to assess the merits of our proposal, LEL-TSK, an experimental study was designed and carried out. This study consists in the application of our method and the statistical methods described above to the forecasting of 23 financial time series. These are 23 daily series from the Dow Jones Industrial Average index, ranging from January 3rd, 1995 to December 31st, 2005. The data were obtained from the NYSE TAQ database. Table 1 shows basic information about each series.

Following the usual procedure, aimed at removing nonstationarity and allowing comparison between series, returns were obtained by taking first order differences of the logarithm of each series. As an example, Fig. 3 shows the effect of the transformation on one of the series.

In order to work under stationary conditions, and to remove high scale variability, the *log-diff* transformation was applied to the 23 series included in Table 1. As seen before, Fig. 3 shows an example of such transformation. Once the series were transformed, each one was split into two periods in order to perform training and testing, respectively. Testing period started in January 3rd, 2005, i.e. the last year was left for testing and the previous 10 years were used for training.

By comparing the Bayesian Information Criterion (BIC) of linear models including all different combination of lagged variables up to order 10, we found that for all the series the highest relevant time step was 4. Hence, the first four past values were used to predict the actual values of the series, i.e. the input vectors for the models were shaped as $(y_{t-4}, y_{t-3}, y_{t-2}, y_{t-1})$.

Table 1
Full name and industry of each of the considered series.

Code	Full name	Industry
aa	Alcoa	Aluminium
aig	American international group	Insurances
axp	American express	Consumer finance
ba	Boeing	Aerospace & defense
cat	Caterpillar	Construction & Mining equipment
dd	DuPont	Chemical industry
dis	Walt Disney	Broadcasting & Entertainment
ge	General electric	Conglomerate
gm	General motors	Automotive
hd	The home depot	Home improvement retailer
hom	Home solutions of America	Construction
ibm	IBM	Computers & technology
jnj	Johnson & Johnson	Pharmaceuticals
jpm	JPMorgan chase	Banking
ko	Coca-Cola	Beverages
mo	Altria group	Tobacco
mcd	McDonald's	Fast food
mmm	3 M	Conglomerate
mrk	Merck	Pharmaceuticals
pfe	Pfizer	Pharmaceuticals
pg	Procter & Gamble	Consumer goods
utx	United technologies corporation	Conglomerate
wmt	Wal-Mart	Retail

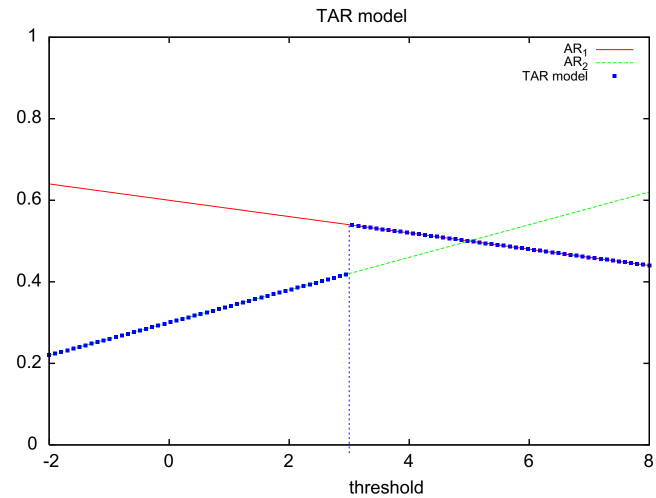

Fig. 1. An example of TAR model.

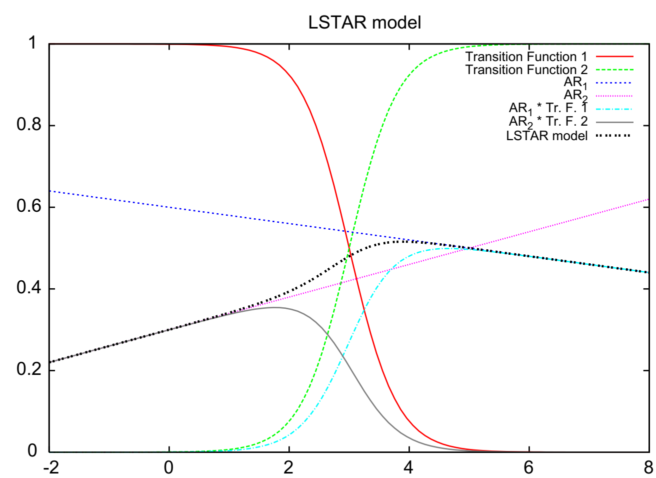

Fig. 2. An example of 2 regime STAR model using logistic transition function.

Once the datasets were built, we trained and applied the models AR, STAR, NCSTAR and LEL-TSK. To compare the one step ahead forecast results, two error measures were chosen: the root mean squared error (RMSE) and the mean absolute error (MAE). If e_i is the difference between the expected value and the value predicted by the models and T is the total number of samples, these common measures are defined as

$$RMSE = \sqrt{\frac{\sum (e_i)^2}{T}} \tag{15}$$

and

$$MAE = \frac{1}{T} \sum |e_i|. \tag{16}$$

Tables 2 and 3 show the results of the experiments. The column *#p* is the number of parameters used by each method, which can be used as a measure of the complexity of the model. Accuracy of methods is shown in the last four columns. For both measures (MAE and RMSE), the first two columns show the in-sample adjustment errors and the last two columns show the forecast on testing

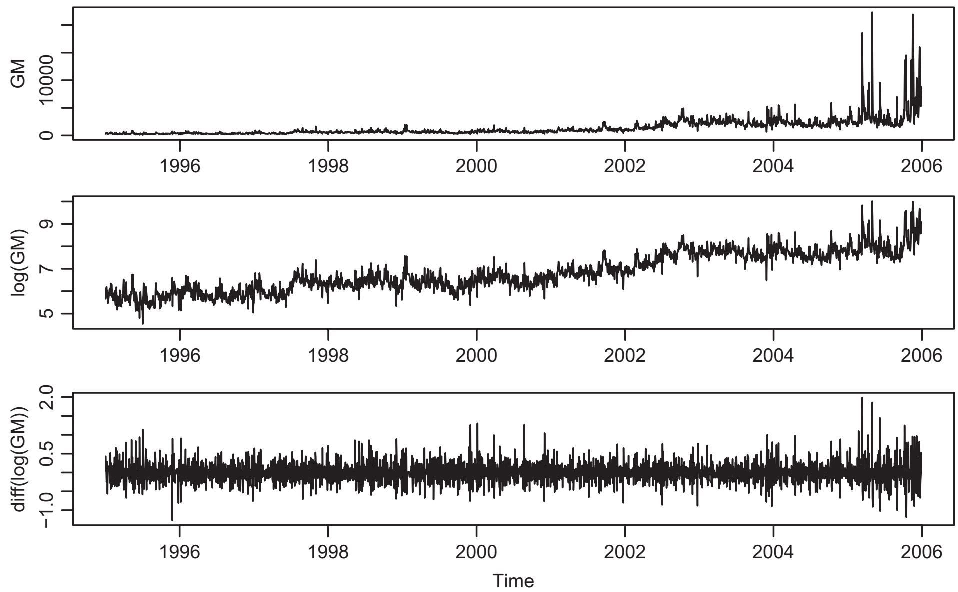

Fig. 3. General motors series with log and difference transformations.

data. The data sets are sorted alphabetically, while models are shown ordered by complexity. On each comparable group the best result is shown in boldface type.

In most series, NCSTAR achieves the best adjustment to training data, followed by STAR and LEL-TSK, while AR performs the worst adjustment in all series. On the other hand, LEL-TSK leads prediction, achieving the best results for all series on testing data. The results are very similar using RMSE or MAE error measures.

Next, the significance of these results is assessed by using statistical tests. To choose the right statistical test, two features of results are noted. First, error rates among the time series considered are not commensurable, as results on different series are not directly comparable. And second, four methods are compared at the same time. Since we are performing multiple comparisons, we can not simply repeat – many times – the tests designed for a pair of variables, as the number of null hypotheses rejected by random chance would become high. Following the methodology recommended by Demsar (2006) for this type of comparisons, we have used Friedman and Nemenyi statistical tests. Friedman test checks if there are significant differences in the group of results, while Nemenyi is used to detect which of all the comparable pairs have significative differences. A detailed description of these tests can be found in Zar's book (Zar, 1999).

To apply these tests, for each time series considered, the methods are ranked according to their results. The best method is assigned a 1, the worst method a 4, and decimal values are applied if ties are found. These rankings are commensurable, so the average can be used to compare the methods.

As there are 4 factors in comparison and the number of samples is $23 \gg 8$ the Chisquare approximation for Friedman statistic is used. The test is performed for training and testing results using both RMSE and MAE measures. In all cases, the null hypothesis (all methods perform equal on the time series considered) is rejected with $p < 0.001$—actually, $p \ll 0.001$.

To know which methods can be considered to perform significantly better than others the Nemenyi critical distance is used. For the comparisons considered, the Nemenyi critical distance is 0.978 at a $p = 0.05$ significance level.

Tables 4 and 5 show the average ranking results for each method on adjustment and testing respectively. The ranking is shown for both measures (RMSE and MAE).

Figs. 4 and 5 represent graphically the relative ranking results of all methods, again, on adjustment and testing, respectively. Each method is shown by a symbol: AR (square), STAR (triangle), NCSTAR (diamond) and LEL-TSK (circle). These symbols are placed (centered) on the *x*-axis according to the average ranking. Top row shows results when using MAE error measure, while bottom row show those using RMSE.

The big rectangle surrounding the ranking line is as wide as the Nemenyi critical distance. Starting on the first method (according to the average ranking), the rectangle includes inside those methods which cannot be regarded as significantly different to the first one. At the same time, it gives an intuitive approximation of which methods are separated by a distance lower or higher than Nemenyi critical distance at 0.05 significance level.

Although the ranking values are different, the significance of results at $p = 0.05$ is the same for MAE and RMSE. This happens for both adjustment and testing. On adjustment, there are significative differences on the following pairs: (NCSTAR, AR) (STAR, AR) (LEL-TSK, AR). On testing, the results are very different from those of adjustment, with the following pairs having significative differences: (LEL-TSK, AR), (LEL-TSK, STAR), (LEL-TSK, NCSTAR), (AR, NCSTAR).

The results show how AR, being the simplest method, is significantly the worst on adjustment. However, AR is the second predicting on test data and significantly better than NCSTAR. The fact that NCSTAR and STAR have been the best on adjustment and the worst on prediction means they have entered into over learning of the adjustment data. This have made them lose generalization and their models are not interesting for these time series, specially compared with a simpler method like AR.

LEL-TSK looks worse on adjustment than STAR or NCSTAR but, actually, no significative difference has been found. Moreover, LEL-TSK has adjusted to training data significantly better than AR. Together with the fact that this method is – significantly – the first on prediction results, we can conclude that LEL-TSK have modeled the behaviour of these time series better than all the others.

The proposed model (LEL-TSK) has a higher number of degrees of freedom and its results have proven to be better. The high complexity of financial time series, as mentioned above, together with the good results obtained by LEL-TSK justifies in this case the choice of a complex model.

Table 2
Results of the experiments for each time series.

	Model	#p	In-sample		Forecast	
			RMSE	MAE	RMSE	MAE
aa	AR	5	0.25029	0.18469	0.22919	0.17358
	STAR	12	0.24802	**0.18272**	0.23436	0.17457
	NCSTAR	27	**0.24586**	0.18158	0.23471	0.17561
	LEL-TSK	170	0.24864	0.18368	**0.20144**	**0.15084**
aig	AR	5	0.21812	0.15737	0.28433	0.20596
	STAR	12	0.21454	0.15505	0.29081	0.20551
	NCSTAR	27	**0.21327**	**0.15419**	0.28865	0.20640
	LEL-TSK	136	0.21551	0.15561	**0.25445**	**0.17866**
axp	AR	5	0.24371	0.17627	0.27766	0.19330
	STAR	12	0.23993	0.17380	0.28943	0.19492
	NCSTAR	27	**0.23883**	**0.17311**	0.29947	0.19879
	LEL-TSK	153	0.24028	0.17443	**0.23470**	**0.16202**
ba	AR	5	0.27051	0.19417	0.27888	0.21595
	STAR	12	0.26919	0.19361	0.28057	0.21646
	NCSTAR	27	**0.26732**	**0.19199**	0.28227	0.21646
	LEL-TSK	136	0.26990	0.19417	**0.24570**	**0.19320**
cat	AR	5	0.25017	0.17994	0.28943	0.21492
	STAR	12	0.24692	**0.17778**	0.29456	0.21905
	NCSTAR	16	0.24746	0.17887	0.29251	0.21703
	LEL-TSK	187	**0.24654**	0.17813	**0.26038**	**0.19293**
dd	AR	5	0.21565	0.15290	0.22508	0.16830
	STAR	12	0.21101	0.14995	0.22637	0.16731
	NCSTAR	16	0.21197	0.15107	0.22532	0.16710
	LEL-TSK	204	**0.21070**	**0.14980**	**0.19621**	**0.14384**
dis	AR	5	0.23517	0.16732	0.19780	0.15081
	STAR	12	**0.23246**	**0.16539**	0.19924	0.15123
	NCSTAR	16	0.23279	0.16541	0.19988	0.15200
	LEL-TSK	187	0.23330	0.16600	**0.18455**	**0.13591**
ge	AR	5	0.21384	0.15344	0.17715	0.12627
	STAR	12	**0.20886**	**0.14995**	0.17902	0.12729
	NCSTAR	16	0.20986	0.15105	0.17874	0.12652
	LEL-TSK	170	0.21028	0.15161	**0.16275**	**0.11238**
gm	AR	5	0.23439	0.17583	0.44652	0.32351
	STAR	12	0.23067	0.17327	0.49008	0.34306
	NCSTAR	27	**0.22857**	**0.17188**	0.51074	0.35422
	LEL-TSK	85	0.23316	0.17532	**0.40861**	0.29093
hd	AR	5	0.24335	0.17248	0.23078	0.16902
	STAR	12	0.23870	0.16991	0.23232	0.16958
	NCSTAR	16	0.23898	0.17059	0.23514	0.17087
	LEL-TSK	204	**0.23788**	**0.16918**	**0.20288**	**0.15204**
hom	AR	5	0.27323	0.19600	0.24765	0.18700
	STAR	12	0.27095	0.19518	0.25016	0.18787
	NCSTAR	49	**0.26516**	**0.19101**	0.25577	0.19402
	LEL-TSK	136	0.27055	0.19456	**0.21538**	**0.15843**
ibm	AR	5	0.27818	0.20079	0.22744	0.17126
	STAR	12	0.27633	0.19989	0.22843	0.17142
	NCSTAR	27	**0.27358**	**0.19805**	0.23098	0.17433
	LEL-TSK	238	0.27609	0.19987	**0.19642**	**0.14566**

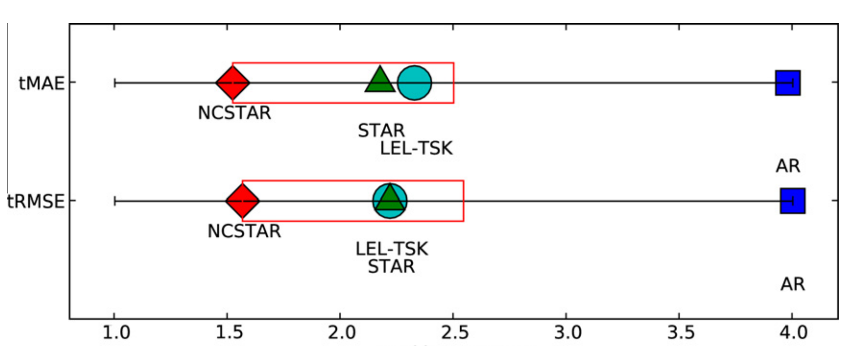

Fig. 4. Comparison of models using ranking in training adjustment.

Table 3
Results of the experiments for each time series.

	Model	#p	In-sample		Forecast	
			RMSE	MAE	RMSE	MAE
jnj	AR	5	0.21613	0.15359	0.19217	0.14715
	STAR	12	0.21326	0.15147	0.19375	0.14742
	NCSTAR	16	**0.21216**	**0.15059**	0.19734	0.14937
	LEL-TSK	136	0.21423	0.15225	**0.16834**	**0.12733**
jpm	AR	5	0.24773	0.18352	0.18275	0.14180
	STAR	12	0.24473	0.18192	0.18073	0.13927
	NCSTAR	27	**0.24185**	**0.17973**	0.18111	0.13930
	LEL-TSK	204	0.24348	0.18092	**0.15747**	**0.12030**
ko	AR	5	0.21603	0.15388	0.18573	0.14156
	STAR	12	0.21273	0.15162	0.18599	0.14108
	NCSTAR	16	0.21332	0.15232	0.18635	0.14206
	LEL-TSK	187	**0.21141**	**0.15104**	**0.16830**	**0.12858**
mo	AR	5	0.28067	0.19822	0.30119	0.20740
	STAR	12	0.27701	0.19522	0.32427	0.21815
	NCSTAR	27	**0.27196**	**0.19184**	0.35093	0.22437
	LEL-TSK	153	0.27710	0.19516	**0.27593**	**0.18622**
mcd	AR	5	0.23092	0.16395	0.26263	0.20139
	STAR	12	0.22833	0.16237	0.26545	0.20198
	NCSTAR	16	**0.22727**	**0.16121**	0.26949	0.20528
	LEL-TSK	170	0.22759	0.16201	**0.22666**	**0.17147**
mmm	AR	5	0.22503	0.16142	0.22207	0.15887
	STAR	12	0.22203	0.15930	0.22318	0.15851
	NCSTAR	38	**0.21732**	**0.15597**	0.22496	0.16390
	LEL-TSK	221	0.21983	0.15742	**0.18844**	**0.13260**
mrk	AR	5	0.24069	0.16821	0.34647	0.20793
	STAR	12	0.23735	0.16564	0.36823	0.21779
	NCSTAR	27	**0.23539**	**0.16410**	0.37859	0.22810
	LEL-TSK	153	0.23743	0.16625	**0.32765**	**0.19806**
pfe	AR	5	0.22455	0.15939	0.26157	0.17798
	STAR	12	0.22035	0.15721	0.27350	0.18280
	NCSTAR	16	**0.22014**	**0.15689**	0.27317	0.18344
	LEL-TSK	119	0.22150	0.15746	**0.24318**	**0.16102**
pg	AR	5	0.22007	0.15156	0.21407	0.15705
	STAR	12	0.21624	0.14877	0.21279	0.15501
	NCSTAR	49	**0.21052**	**0.14548**	0.21690	0.16122
	LEL-TSK	170	0.21594	0.14896	**0.19135**	**0.14161**
utx	AR	5	0.23811	0.17443	0.22387	0.17384
	STAR	12	0.23638	0.17292	0.22078	0.17128
	NCSTAR	16	0.23580	**0.17224**	0.21929	0.16984
	LEL-TSK	238	**0.23519**	0.17289	**0.18915**	**0.14869**
wmt	AR	5	0.22748	0.16700	0.23202	0.17630
	STAR	12	0.22306	0.16487	0.23711	0.17769
	NCSTAR	16	0.22384	0.16565	0.23614	0.17711
	LEL-TSK	119	**0.22278**	**0.16463**	**0.20316**	**0.15378**

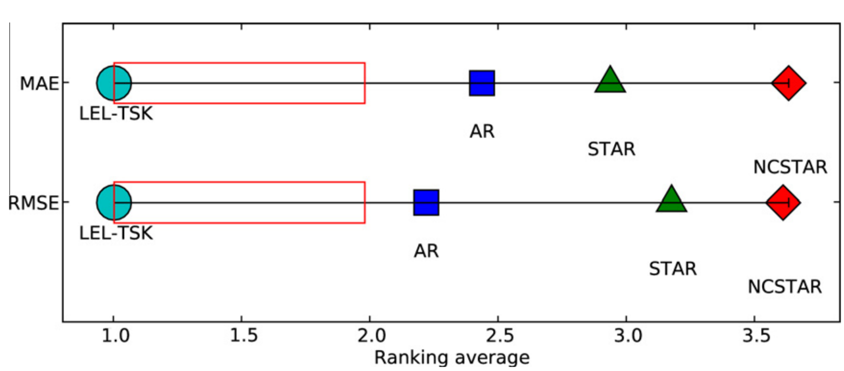

Fig. 5. Comparison of models using ranking on test data.

Table 4
Averages of method ranking in training adjustment.

Model	Avg. ranking for RMSE	Avg. ranking for MAE
AR	4.00	3.98
STAR	2.22	2.17
NCSTAR	1.57	1.52
LEL-TSK	2.22	2.33

Nemenyi critical distance (p = 0.05): 0.978.

Table 5
Averages of method ranking on test data.

Model	Avg. ranking for RMSE	Avg. ranking for MAE
AR	2.22	2.43
STAR	3.17	2.93
NCSTAR	3.61	3.63
LEL-TSK	1.00	1.00

Nemenyi critical distance (p = 0.05): 0.978.

The problem of predicting financial time series is already considered a complex problem, so it is logical to think that a more complex system may suit them better. In this way, the models evolved by LEL-TSK are more adequate to predict stock price time series than the others compared in this work.

5. Conclusions

We have addressed the problem of financial time series forecasting from a Soft Computing perspective. This is a complex problem which usually requires the generation of accurate non-linear models. We propose a solution based on the evolution of fuzzy rule-based models through an effective evolutionary algorithm, LEL-TSK.

The performance of the proposed model is compared to that of a number of statistical methods. A thorough empirical analysis has been designed and carried out. This study includes the modeling of 23 daily series from the Dow Jones Industrial Average index. The results state that our proposal consistently outperformed the other considered methods. This conclusion can be further asserted since the statitiscal tests applied make statistically significant the differences in performance shown.

While the supremacy in performance has been clearly established our next goal is to increase the interpretability of the constructed systems by decreasing their complexity. New work in this line has already begun.

Acknowledgments

The research included in this work has been partially supported by the Ministerio de Ciencia e Innovación through Research Grant Ref. TIN2009-14575.

References

Alcalá, R., Alcalá-Fdez, J., Casillas, J., Cordón, O., & Herrera, F. (2007). Local identification of prototypes for genetic learning of accurate TSK fuzzy rule-based systems. *International Journal of Intelligent Systems, 22*(9), 909–941.

Armano, G., Marchesi, M., & Murru, A. (2005). A hybrid genetic-neural architecture for stock indexes forecasting. *Information Sciences, 170*, 3–33.

Box, G. E. P., & Jenkins, G. M. (1970). *Time series analysis: Forecasting and control*. San Francisco: Holden–Day.

Chen, R., & Tsay, R. (1993). Functional-coefficient autoregressive models. *Journal of the American Statistical Association, 88*, 298–308.

Chen, Y., Yuan, B., Dong, J., & Abraham, A. (2005). Time-series forecasting using flexible neural tree model. *Information Sciences, 174*, 219–235.

Cordón, O., del Jesus, M., Herrera, F., & Lozano, M. (1999). Mogul: A methodology to obtain genetic fuzzy rule-based systems under the iterative rule learning approach. *International Journal of Intelligent Systems, 14*(9), 1123–1153.

Cordón, O., & Herrera, F. (1999). A two-stage evolutionary process for designing tsk fuzzy rule-based systems. *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics, 29*(6), 703–715.

Cordón, O., & Herrera, F. (2001). Hybridizing genetic algorithms with sharing scheme and evolution strategies for designing approximate fuzzy rule-based systems. *Fuzzy Sets and Systems, 118*(2), 235–255.

da Silva, C. (2008). Time series forecasting with a non-linear model and the scatter search meta-heuristic. *Information Sciences, 178*, 3288–3299.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research, 7*, 1–30.

Hellstrom, T., & Holmstrom, K. (1998). Predicting the stock market, Tech. rep., Malardalen University. <http://www.cs.umu.se/thomash/reports/pred.pdf>.

Medeiros, M., & Veiga, A. (2005). A flexible coefficient smooth transition time series model. *IEEE Transactions on Neural Networks, 16*(1), 97–113.

Pissarenko, D. (2002). Neural networks for financial time series prediction: Overview over recent research.

Rivas, V., Merelo, J., Castillo, P., Arenas, M., & Castellano, J. (2004). Evolving rbf neural networks for time-series forecasting with evrbf. *Information Sciences, 165*, 207–220.

Sheta, A., & Jong, K. D. (2001). Time-series forecasting using ga-tune radial basis functions. *Information Sciences, 133*, 221–228.

Suarez-Farinas, M., Pedreira, C. E., & Medeiros, M. C. (2004). Local global neural networks: A new approach for nonlinear time series modeling. *Journal of the American Statistical Association, 99*, 1092–1107. URL <http://ideas.repec.org/a/bes/jnlasa/v99y2004p1092-1107.html>.

Sugeno, M., & Kang, G. (1988). Structure identification of fuzzy model. *Fuzzy Sets and Systems, 28*, 15–33.

Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems Man and Cybernetics, 15*, 116–132.

Tang, Y., & Xu, Y. (2005). Application of fuzzy naive bayes and a real-valued genetic algorithm in identification of fuzzy model. *Information Sciences, 169*, 205–226.

Teräsvirta, T. (1994). Specification, estimation and evaluation of smooth transition autoregresive models. *Journal of the American Statistical Association, 89*, 208–218.

Tong, Howell (1978). On a threshold model. In C. Chen (Ed.), *Pattern recognition and signal processing. NATO ASI Series E: Applied Sec. No. 29* (pp. 575-586). Netherlands: Sijthoff & Noordhoff.

van Dijk, D., Teräsvirta, T., & Franses, P. H. (2002). Smooth transition autoregresive models – A survey of recent developments. *Econometric Reviews, 21*(1), 1–47.

Xia, Y., & Li, W. (1999). On single-index coefficient regression models. *Journal of the American Statistical Association, 94*, 1275–1285.

Zadeh, L. (1965). Fuzzy sets. *Information and control, 3*(8), 338–353.

Zar, J. H. (1999). *Biostatistical analysis* (4th ed.). New Jersey (US): Prentice-Hall.

Zhang, G. (2007). A neural network ensemble method for jittered training data for time series forecasting. *Information Sciences, 177*, 5329–5346.