

# Intelligent Systems in Long-Term Forecasting of the Extra-Virgin Olive Oil Price in the Spanish Market

María Dolores Pérez-Godoy<sup>1</sup>, Pedro Pérez<sup>1</sup>, Antonio Jesús Rivera<sup>1</sup>,  
María José del Jesús<sup>1</sup>, María Pilar Frías<sup>2</sup>, and Manuel Parras<sup>3</sup>

<sup>1</sup> Department of Computer Science

<sup>2</sup> Department of Statistics and Operation Research

<sup>3</sup> Department of Marketing University of Jaén, Spain

{lperez,arivera,mjjesus}@ujaen.es, pedro.perez.recuerta@gmail.com,  
mpfrias@ujaen.es, mparras@ujaen.es

**Abstract.** In this paper the problem of estimating forecasts, for the Official Market of future contracts for olive oil in Spain, is addressed. Time series analysis and their applications is an emerging research line in the Intelligent Systems field. Among the reasons for carry out time series analysis and forecasting, the associated increment in the benefits of the implied organizations must be highlighted. In this paper an adaptation of CO<sup>2</sup>RBFN, evolutionary COoperative-COmpetitive algorithm for Radial Basis Function Networks design, applied to the long-term prediction of the extra-virgin olive oil price is presented. This long-term horizon has been fixed to six months. The results of CO<sup>2</sup>RBFN have been compared with other data mining methods, typically used in time series forecasting, such as other neural networks models, a support vector machine method and a fuzzy system.

## 1 Introduction

Spatial and/or temporal data mining [13][22] is a growing application task inside the intelligent systems (data mining methods) field. There are different reasons for this growth, such as: the increase of this kind of data which are collected and requiring analysis, their availability in Internet, the commercial advantage of the results obtained, etc.

Time series analysis and forecasting is one of the most important research topic of temporal data mining. A time series is a set of regular time-ordered observations of a quantitative characteristic of an individual phenomenon taken at successive periods or points of time. The problems in which the data are not independent but also have a temporal relationship are called time series forecasting problems. The ultimate aim in these problems is to increase our knowledge of a phenomenon or aspect in order to have a better understanding of it in the future.

Olive oil has become an important business sector in a continuously expanding market. In 2009 (<http://www.mfao.es>), World produced 2,888,000 of tons of olive

oil, Spain is the first olive oil producing and exporting country and Jaén is the most productive province of Spain, with 430,000 tons, the 15% of the total production in the planet.

This is especially important in the Official Market for the negotiation of Future contracts for Olive Oil (MFAO) in Spain: a society whose objective is to discover prices that will balance, supply and demand, in some future time periods. Our aim is to predict these future prices in order to increase the global benefits of the sector.

The time series forecasting problem is usually addressed with data mining methods, such as neural networks [3][23][21][7] or fuzzy rule based systems [2][14][15][27]. Moreover statistic models, such as ARIMA [4], are used in time series analysis but this kind of methods are not suitable in long-term predictions [9].

Radial Basis Function Networks (RBFNs) are an important artificial neural network paradigm [5] with interesting characteristics such as a simple topological structure or universal approximation ability [18]. The overall efficiency of RBFNs has been proved in many areas such as pattern classification [6], function approximation [18] or time series prediction [25]. Typically these networks are design by means of evolutionary algorithms [11].

Authors have developed an evolutionary cooperative-competitive method, CO<sup>2</sup>RBFN [19], for the design of RBFNs. An adaptation of this algorithm has proven its efficiency in short-term predictions [20]. The objective of this paper is to test the efficiency of the adapted CO<sup>2</sup>RBFN in the long-term forecasting of the extra-virgin olive oil price in the Spanish market. It must be highlighted that, for first time, an horizon of six months has been fixed for this problem.

The results obtained using CO<sup>2</sup>RBFN are also compared with other intelligent systems methods, typically used in time series forecasting, such a classical method for developing Fuzzy Systems (Fuzzy-WM) [24], a MultiLayer Perceptron Network trained using a Conjugate Gradient learning algorithm (MLP-ConjGrad) [17], a support vector machine (NU-SVR) [8], and a classical design method for Radial Basis Function Network learning (RBFN-LMS) [26].

This paper is organized as follows: section 2 describes CO<sup>2</sup>RBFN and its extension to time series forecasting. In section 3 the other forecasting methods are described. The study and results obtained for the forecast methods are detailed in Section 4. In Section 5, conclusions and future works are outlined.

## 2 CO<sup>2</sup>RBFN for Time Series Forecasting

CO<sup>2</sup>RBFN [19], is an hybrid evolutionary cooperative-competitive algorithm for the design of RBFNs. In this algorithm each individual of the population represents, with a real representation, an RBF and the entire population is responsible for the final solution. The individuals cooperate towards a definitive solution, but they must also compete for survival. In this environment, in which the solution depends on the behavior of many components, the fitness of each individual is known as credit assignment. In order to measure the credit assignment of an

individual, three factors have been proposed: the RBF contribution to the network output, the error in the basis function radius, and the degree of overlapping among RBFs.

The application of the operators is determined by a Fuzzy Rule-Based System. The inputs of this system are the three parameters used for credit assignment and the outputs are the operators' application probability.

The main steps of CO<sup>2</sup>RBFN, explained in the following subsections, are shown in the pseudocode in Figure 1.

1. Initialize RBFN
2. Train RBFN
3. Evaluate RBFs
4. Apply operators to RBFs
5. Substitute the eliminated RBFs
6. Select the best RBFs
7. If the stop condition is not verified go to step 2

**Fig. 1.** Main steps of CO<sup>2</sup>RBFN

**RBFN initialization.** To define the initial network a specified number  $m$  of neurons (i.e. the size of population) is randomly allocated among the different patterns of the training set. To do so, each RBF centre,  $\mathbf{c}_i$ , is randomly established to a pattern of the training set. The RBF widths,  $d_i$ , will be set to half the average distance between the centres. Finally, the RBF weights,  $w_{ij}$ , are set to zero.

**RBFN training.** The Least Mean Square algorithm [26] has been used to calculate the RBF weights.

**RBF evaluation.** A credit assignment mechanism is required in order to evaluate the role of each RBF  $\phi_i$  in the cooperative-competitive environment. For an RBF, three parameters,  $a_i, e_i, o_i$  are defined:

- The contribution,  $a_i$ , of the RBF  $\phi_i, i = 1 \dots m$ , is determined by considering the weight,  $w_i$ , and the number of patterns of the training set inside its width,  $np_i$ . An RBF with a low weight and few patterns inside its width will have a low contribution:

$$a_i = \begin{cases} |w_i| & \text{if } pi_i > q \\ |w_i| * (np_i/q) & \text{otherwise} \end{cases} \tag{1}$$

where  $q$  is the average of the  $np_i$  values minus the standard deviation of the  $np_i$  values.

- The error measure,  $e_i$ , for each RBF  $\phi_i$ , is obtained by calculating the Mean Absolute Percentage Error (MAPE) inside its width:

$$e_i = \frac{\sum_{\forall pi_i} \left| \frac{f(pi_i) - y(pi_i)}{f(pi_i)} \right|}{np_i} \tag{2}$$

where  $f(pi_i)$  is the output of the model for the point  $pi_i$ , inside the width of RBF  $\phi_i$ ,  $y(pi_i)$  is the real output at the same point, and  $npi_i$  is the number of points inside the width of RBF  $\phi_i$ .

- The overlapping of the RBF  $\phi_i$  and the other RBFs is quantified by using the parameter  $o_i$ . This parameter is computed by taking into account the fitness sharing methodology [10], whose aim is to maintain the diversity in the population. This factor is expressed as:

$$o_i = \sum_{j=1}^m o_{ij} \quad o_{ij} = \begin{cases} (1 - \|\phi_i - \phi_j\|/d_i) & \text{if } \|\phi_i - \phi_j\| < d_i \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where  $o_{ij}$  measures the overlapping of the RBF  $\phi_i$  y  $\phi_j$   $j = 1 \dots m$ .

**Applying operators to RBFs.** In CO<sup>2</sup>RBFN four operators have been defined in order to be applied to the RBFs:

- Operator Remove: eliminates an RBF.
- Operator Random Mutation: modifies the centre and width of an RBF in a percentage below 50% of the old width.
- Operator Biased Mutation: modifies the width and all coordinates of the centre using local information of the RBF environment. In the same way that the LMS algorithm.
- Operator Null: in this case all the parameters of the RBF are maintained.

The operators are applied to the whole population of RBFs. The probability for choosing an operator is determined by means of a Mandani-type fuzzy rule based system [16] which represents expert knowledge about the operator application in order to obtain a simple and accurate RBFN. The inputs of this system are parameters  $a_i$ ,  $e_i$  and  $o_i$  used for defining the credit assignment of the RBF  $\phi_i$ . These inputs are considered as linguistic variables  $va_i$ ,  $ve_i$  and  $vo_i$ . The outputs,  $p_{remove}$ ,  $p_{rm}$ ,  $p_{bm}$  and  $p_{null}$ , represent the probability of applying Remove, Random Mutation, Biased Mutation and Null operators, respectively.

Table 1 shows the rule base used to relate the described antecedents and consequents. In the table each row represents one rule. For example, the interpretation of the first rule is: If the contribution of an RBF is Low Then the probability of applying the operator Remove is Medium-High, the probability of applying the operator Random Mutation is Medium-High, the probability of

**Table 1.** Fuzzy rule base representing expert knowledge in the design of RBFNs

Antecedents			Consequents				Antecedents			Consequents			
$va$	$ve$	$vo$	$p_{remove}$	$p_{rm}$	$p_{bm}$	$p_{null}$	$va$	$ve$	$vo$	$p_{remove}$	$p_{rm}$	$p_{bm}$	$p_{null}$
R1	L		M-H	M-H	L	L	R6	H		M-H	M-H	L	L
R2	M		M-L	M-H	M-L	M-L	R7	L		L	M-H	M-H	M-H
R3	H		L	M-H	M-H	M-H	R8	M		M-L	M-H	M-L	M-L
R4	L	L	L	M-H	M-H	M-H	R9	H		M-H	M-H	L	L
R5	M		M-L	M-H	M-L	M-L							

applying the operator Biased Mutation is Low and the probability of applying the operator null is Low.

**Introduction of new RBFs.** In this step, the eliminated RBFs are substituted by new RBFs. The new RBF is located in the centre of the area with maximum error or in a randomly chosen pattern with a probability of 0.5 respectively. The width of the new RBF will be set to the average of the RBFs in the population plus half of the minimum distance to the nearest RBF. Its weights are set to zero.

**Replacement strategy.** The replacement scheme determines which new RBFs (obtained before the mutation) will be included in the new population. To do so, the role of the mutated RBF in the net is compared with the original one to determine the RBF with the best behaviour in order to include it in the population.

### 3 Other Forecasting Methods

In this section other typically forecasting methods to apply to the long-term forecasting of the extra-virgin olive oil price are presented. These methods are a classical method for developing Fuzzy Systems (Fuzzy-WM)[24], a MultiLayer Perceptron Network trained using a Conjugate Gradient learning algorithm (MLP-ConjGrad)[17], a support vector machine (NU-SVR)[8], and a classical design method for Radial Basis Function Network learning (RBFN-LMS)[26]. Concretely:

- Fuzzy-WM [24]. This fuzzy system based design algorithm generates fuzzy rules from numerical input-output data pairs of the dataset. Domain intervals of the dataset are divided in regions and each region is assigned to a fuzzy membership function. For each input-output data pair one rule is generated. These fuzzy rules will represent a mapping from input to output space.
- MLP-ConjGrad [17]. MLP-ConjGrad uses the conjugate-gradient algorithm to adjust weight values of a multilayer perceptron [12]. Compared to gradient descent, the conjugate gradient algorithm takes a more direct path to the optimal set of weight values. Usually, the conjugate gradient is significantly faster and more robust than the gradient descent.
- RBFN-LMS. It builds an RBFN with a pre-specified number of RBFs. By means of the K-Means clustering algorithm it chooses an equal number of points from the training set to be the centres of the neurons. Then, it establishes a single radius for all the neurons as half the average distance between the set of centres. Finally, weights are analytically computed using the LMS algorithm [26].
- NU-SVR. The SVM (Support Vector Machine) model uses the sequential minimal optimization training algorithm and treats a given problem in terms of solving a quadratic optimization problem. The NU-SVR, called also v-SVM, for regression problems is an extension of the traditional SVM and it aims to build a loss function [8].

### 4 Experimentation and Results

The dataset used in the experimentation have been obtained from *Poolred*, an initiative of the Foundation for the Promotion and Development of the Olive and Olive Oil located in Jaén, Spain (<http://www.oliva.net/poolred/>). The time series dataset contains the monthly extra-virgin olive oil price per ton. With these data the addressed task is to forecast the extra-virgin olive oil price after six months. In this study, the data used are the price at origin from the 1<sup>st</sup> month of the year 2002 to the 12<sup>th</sup> month of the year 2008 in Spain. The cases in the data set were divided into two subsets: one for training and the other for testing. The data from the 1<sup>st</sup> month of 2002 to the 12<sup>th</sup> month of 2008 were used for training. The performance of the different predictions and methods were tested by estimating the data from the 1<sup>st</sup> month to the 12<sup>th</sup> month of 2008. Figure 2 shows the time series data and training and test datasets.

As mentioned, experiments carry out predictions with horizons of six months. In this way the patterns for data mining methods are heuristically (recommended by the experts of the sector) composed of  $(n - 5, n - 4, n - 3, n - 2, n - 1, n, n + 6)$ , when the price to forecast is  $n + 1$  and must be determined from the past prices  $n - 5$  to  $n$ .

The series has been differentiated to avoid problems related with the stationarity. The predictions have been performed using the differenced data, but errors have been calculated after reconstruct the original series.

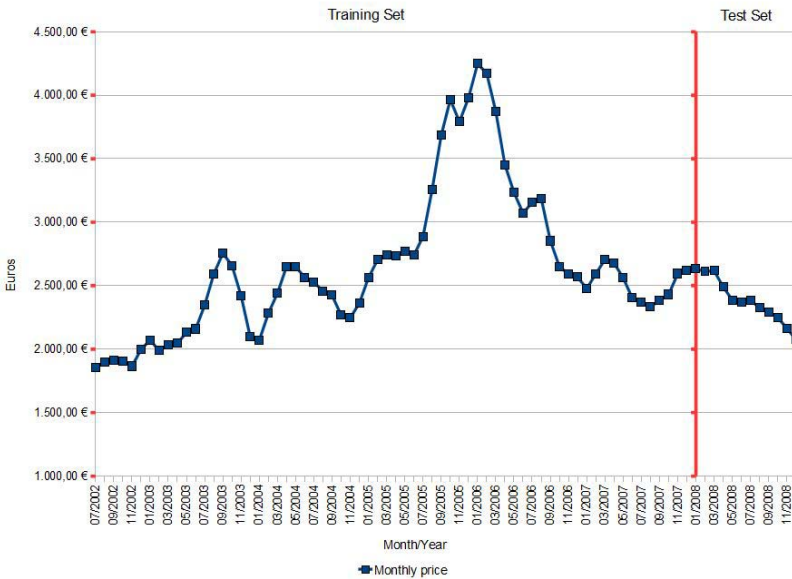


Fig. 2. Monthly extra-virgin olive oil prices in Tons / Euro

The implementation of the data mining methods has obtained from KEEL [1]. The parameters used in these data mining methods are the values recommended in the literature. For CO<sup>2</sup>RBFN the number of executions is 200 and the number of RBFs or individuals in the population is set to 10. To obtain the results, algorithms have been executed 10 times (repetitions).

To estimate prediction capacity, the error considered is the Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{\sum_i^n (|f_i - y_i|/f_i)}{n} \tag{4}$$

where  $f_i$  is the predicted output of the model and  $y_i$  is the desired output.

Table 2 shows the obtained statistics, average and standard deviation for 10 repetitions, for MAPE. NU-SVR and Fuzzy-WM are deterministic methods and obviously the standard deviation for 10 repetitions is 0. Figure 3 shows the prediction achieved in the test set by the best training repetition of the methods.

**Table 2.** Results obtained by different methods forecasting the price of olive oil

Method	Statistics for 10 repetitions	
	Training MAPE	Test MAPE
Fuzzy-WM	0,08520 ± 0,0	0,21378 ± 0,0
MLP-ConjGrad	0,05270 ± 0,02007	0,19958 ± 0,02566
NU-SVR	0,10484 ± 0,0	0,18131 ± 0,0
RBFN-LMS	0,04343 ± 0,01875	0,17789 ± 0,03469
CO <sup>2</sup> RBFN	0,10622 ± 0,00362	0,17567 ± 0,01111

If we analyze the results we can draw the following conclusions:

- The method proposed by the authors, CO<sup>2</sup>RBFN, is the best method in average.
- Moreover, CO<sup>2</sup>RBFN has the lower standard deviation, obviously for non-deterministic, which demonstrates the robustness of the method.
- With only 10 neurons, CO<sup>2</sup>RBFN is the method with lowest complexity, along with MLP-ConjGrad.
- The dataset problem has a bias to overtraining and models such as RBFN-LMS (50 neurons) or MLP-ConjGrad have obtained lower training errors but they suffer overtraining in the test dataset.
- Methods do not achieve good predictions for the latest data in the test set, see Figure 3. Perhaps the behavior or the slope of these data is not in the training set.

Finally, it must be highlighted that the accuracy of the results obtained for this long-term prediction has been of interest for the olive-oil sector experts.

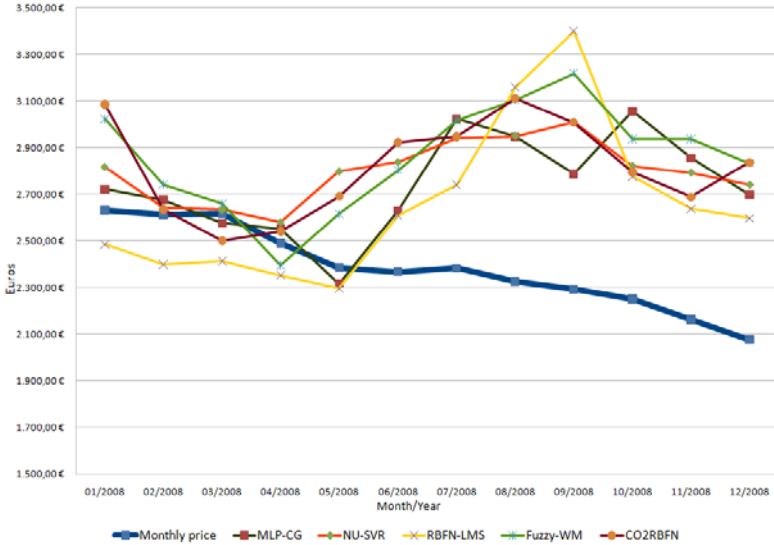


Fig. 3. Forecasting in the test set of the best training repetition

## 5 Concluding Remarks

This paper presents an application of the intelligent systems to the prediction of the extra-virgin olive oil price. This task is very important for the official Market for the negotiation of Futures contracts for Olive Oil (MFAO) in Spain: a society whose objective is to negotiate an appropriate price for the olive oil at the moment it is to be sold at a fixed time in the future. It must be highlighted that for first time the prediction horizon has been fixed to six months, therefore long-term predictions have been estimated.

With this aim in mind authors have tested an adaptation of their evolutionary cooperative-competitive algorithm (CO<sup>2</sup>RBFN) to the forecasting of the extra-virgin olive oil price. As important key point of our proposal it is must be highlighted the identification of the role (credit assignment) of each basis function in the whole network. It is defined by three factors are defined and used: the RBF contribution to the network’s output,  $a_i$ ; the error in the basis function radius,  $e_i$ ; and the degree of overlapping among RBFs,  $o_i$ . Another important key is that the application of the evolutive operators is determined by a fuzzy rule-based system which represents expert knowledge of the RBFN design. The inputs of this system are the three parameters used for credit assignment.

Typically data mining methods have been applied for comparisons such as methods for designing MLP networks, fuzzy systems, support vector machines of RBFNs.

From the results it can be conclude that CO<sup>2</sup>RBFN is the best method in average for this problem. Moreover, the lowest standard deviation of CO<sup>2</sup>RBFN



demonstrates the robustness of the method. Finally, CO<sup>2</sup>RBFN developed the model with the lowest complexity, with only 10 neurons.

As future lines, pre-processing for feature selection and exogenous features like meteorology or econometric data can be taken into account in order to increase the performance of the forecast.

**Acknowledgments.** Supported by the Spanish Ministry of Science and Technology under the Projects TIN2008-06681-C06-02, the Andalusian Research Plan TIC-3928 and the Project of the University of Jaén UJA-08-16-30.

## References

1. Alcalá-Fdez, J., Sánchez, L., García, S., Del Jesus, M.J., Ventura, S., Garrell, J.M., Otero, J., Romero, C., Bacardit, J., Rivas, V., Fernández, J.C., Herrera, F.: KEEL: A Software Tool to Assess Evolutionary Algorithms for Data Mining Problems. *Soft Computing* 13(3), 307–318 (2009)
2. Azadeh, A., Saberi, M., Ghaderi, S.F., Gitiforouz, A., Ebrahimipour, V.: Improved estimation of electricity demand function by integration of fuzzy system and data mining approach. *Energy Conversion and Management* 48(8), 2165–2177 (2008)
3. Aznarte, J.L., Nieto, D., Benítez, J.M., Alba, F., de Linares, C.: Forecasting airborne pollen concentration time series with neural and neuro-fuzzy models. *Expert Systems with Applications* 32, 1218–1225 (2007)
4. Box, G., Jenkins, G.: *Time series analysis: forecasting and control*, revised edn. Holden Day, San Francisco (1976)
5. Broomhead, D., Lowe, D.: Multivariable functional interpolation and adaptive networks. *Complex System* 2, 321–355 (1998)
6. Buchtala, O., Klimek, M., Sick, B.: Evolutionary optimization of radial basis function classifiers for data mining applications. *IEEE Transactions on Systems, Man and Cybernetics Part B* 35(5), 928–947 (2005)
7. Co, H.C., Boosarawongse, R.: Forecasting Thailand's rice export: Statistical techniques vs. artificial neural networks. *Computers and Industrial Engineering* 53(4), 610–627 (2007)
8. Fan, R.E., Chen, P.H., Lin, C.J.: Working set selection using the second order information for training SVM. *Journal of Machine Learning Research* (6), 1889–1918 (2005)
9. Franses, P.H., van Dijk, D.: *Non-linear time series models in empirical finance*. Cambridge University Press, Cambridge (2000)
10. Goldberg, D., Richardson, J.: Genetic algorithms with sharing for multimodal function optimization. In: Grefenstette (ed.) *Proc. Second International Conference on Genetic Algorithms*, pp. 41–49. Lawrence Erlbaum Associates, Mahwah (1987)
11. Harpham, C., Dawson, C.W., Brown, M.R.: A review of genetic algorithms applied to training radial basis function networks. *Neural Computing and Applications* 13, 193–201 (2004)
12. Haykin, S.: *Neural Networks: A Comprehensive Foundation*, 2nd edn. Prentice-Hall, Englewood Cliffs (1998)
13. Hsu, W., Li Lee, M., Wang, J.: *Temporal and Spatio-temporal Data Mining*. IGI Publishing (2007)
14. Jang, J.R.: ANFIS: Adaptive-Network-based Fuzzy Inference System. *IEEE Trans. Systems, Man and Cybernetics* 23(3), 665–685 (1993)

15. Khashei, M., Reza Hejazi, S., Bijari, M.: A new hybrid artificial neural networks and fuzzy regression model for time series forecasting. *Fuzzy Sets and Systems* 159(7), 769–786 (2008)
16. Mandani, E., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine* 7(1), 1–13 (1975)
17. Moller, F.: A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks* 6, 525–533 (1990)
18. Park, J., Sandberg, I.: Universal approximation using radial-basis function networks. *Neural Computation* 3, 246–257 (1991)
19. Pérez-Godoy, M.D., Rivera, A.J., Berlanga, F.J., del Jesus, M.J.: CO2RBFN: an evolutionary cooperative-competitive RBFN design algorithm for classification problems. *Soft Computing* (2009) doi: 10.1007/s00500-009-0488-z
20. Pérez-Godoy, M.D., Pérez, P., Rivera, A.J., del Jesus, M.J., Frías, M.P., Parras, M.: CO2RBFN for short-term forecasting of the extra-virgin olive oil price in the Spanish market. *International Journal of Hybrid Intelligent Systems* 7(1), 75–87 (2010)
21. Pino, R., Parreno, J., Gomez, A., Priore, P.: Forecasting next-day price of electricity in the Spanish energy market using artificial neural networks. *Engineering Applications of Artificial Intelligence* 21(1), 53–62 (2008)
22. Roddick, J.F., Spiliopoulou, M.: A bibliography of temporal, spatial and spatio-temporal data mining research. *ACM SIGKDD Explorations Newsletter* 1(1), 34–38 (1999)
23. Ture, M., Kurt, I.: Comparison of four different time series methods to forecast hepatitis A virus infection. *Expert Systems with Applications* 31(1), 41–46 (2006)
24. Wang, L.X., Mendel, J.M.: Generating Fuzzy Rules by Learning from Examples. *IEEE Transactions on Systems, Man and Cybernetics* 22(6), 1414–1427 (1992)
25. Whitehead, B., Choate, T.: Cooperative-competitive genetic evolution of Radial Basis Function centers and widths for time series prediction. *IEEE Trans. on Neural Networks* 7(4), 869–880 (1996)
26. Widrow, B., Lehr, M.A.: 30 Years of adaptive neural networks: perceptron, madaline and backpropagation. *Proceedings of the IEEE* 78(9), 1415–1442 (1990)
27. Yu, T., Wilkinson, D.: A co-evolutionary fuzzy system for reservoir well logs interpretation. *Evolutionary computation in practice*, 199–218 (2008)