Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Food Engineering 75 (2006) 196–204
JOURNAL OF FOOD ENGINEERING
www.elsevier.com/locate/jfoodeng

Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing

Philip Doganis, Alex Alexandridis, Panagiotis Patrinos, Haralambos Sarimveis *

School of Chemical Engineering, National Technical University of Athens, 9 Heroon Polytechniou Street, Zografou Campus, Athens 15780, Greece

Received 29 September 2003; received in revised form 3 March 2005; accepted 7 March 2005
Available online 21 June 2005

Abstract

Due to the strong competition that exists today, most manufacturing organizations are in a continuous effort for increasing their profits and reducing their costs. Accurate sales forecasting is certainly an inexpensive way to meet the aforementioned goals, since this leads to improved customer service, reduced lost sales and product returns and more efficient production planning. Especially for the food industry, successful sales forecasting systems can be very beneficial, due to the short shelf-life of many food products and the importance of the product quality which is closely related to human health. In this paper we present a complete framework that can be used for developing nonlinear time series sales forecasting models. The method is a combination of two artificial intelligence technologies, namely the radial basis function (RBF) neural network architecture and a specially designed genetic algorithm (GA). The methodology is applied successfully to sales data of fresh milk provided by a major manufacturing company of dairy products.

Keywords: Sales forecasting; Dairy products; Fresh milk; Neural networks; Evolutionary computation; Genetic algorithms

1. Introduction

In today's strong competition, a key issue that defines the success of a manufacturing organization is its ability to adapt easily to the changes of its business environment. To this end, it is very useful for a modern company to have a good estimate of how key indicators are going to behave in the future, a task that is fulfilled by forecasting. An efficient forecasting system can improve machine utilization, reduce inventories, achieve greater flexibility to changes and increase profits. In particular, sales forecasting is very important, as its outcome is used by many functions in the organization (Mentzer & Bienstock, 1998): Finance and accounting departments are able to project cost, profit levels and capital needs based on a sales forecast. The sales department requires a good knowledge of the sales volume of each product, as it is charged with the job of organizing the sales force. Production/purchasing needs a long-term forecast for planning the development of the plant and equipment and a more detailed short-term forecast for arranging the production plan. Marketing needs a view of the future market in order to plan its actions and assess the impact of changes in the marketing strategy on sales volumes. Finally, logistics also needs accurate sales forecasts of different horizon lengths: a long-term forecast in order to develop and organize logistics infrastructure and a short-term forecast to define specific logistics needs.

Food companies are more concerned with sales forecasting due to their special characteristics, such as the short shelf-life of their products, the need to maintain high product quality and the uncertainty and fluctuations in consumer demands. As products can only be

* Corresponding author. Tel.: +30 210 7723237; fax: +30 210 7723138.
E-mail address: hsarimv@chemeng.ntua.gr (H. Sarimveis).

0260-8774/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jfoodeng.2005.03.056

sold for a limited period of time, both shortage and surplus of goods can lead to loss of income for the company. The variations in consumer demand are caused by factors like price, promotions, changing consumer preferences or weather changes (Van der Vorst, Beulens, De Wit, & Van Beek, 1998). A recent initiative of several large companies in the food industry, which aimed to improve forecasting practice, identified that 48% of food companies are poor at forecasting (Adebanjo & Mann, 2000).

The methodologies that have been used in sales forecasting are typically time series algorithms that can be classified as linear or nonlinear, depending on the nature of the model they are based on. Linear models, like autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) (Box, Jenkins, & Reinsel, 1994) are the most popular methodologies, but their forecasting ability is limited by their assumption of a linear behavior and thus, it is not always satisfactory (Zhang, 2003). In order to address possible nonlinearities in time series modeling, researchers introduced a number of nonlinear methodologies, including nonlinear ARMA time series models. Their main drawback is that the type of nonlinearity is not known in advance and the modeler needs to select the structure of the model by trial and error. Advanced artificial intelligence technologies, like artificial neural networks (ANN) (Haykin, 1994) and fuzzy logic systems use more sophisticated generic model structures that can incorporate the characteristics of complex data and produce accurate time series models, by eliminating the time consuming trial and error procedure.

Obviously, the key question concerns the accuracy of each modeling method. To this end, a number of studies have been conducted to compare the aforementioned methods and as we will show in the sequel, the results are not clearly in favor of one particular method. ANNs have been applied successfully to problems concerning sales of food products (peanut butter and ketchup), such as predicting the impact of promotional activities and consumer choice on the sales volumes at a retail store (Agrawal & Schorling, 1996) and were found to perform better than linear models. Another interesting example is by Ainscough and Aronson (1999) that compared ANNs to linear regression for studying the same effects on yogurt. Zhang, Patuwo, and Hu (1998) did a comprehensive review of the literature concerning the utilization of ANNs in forecasting problems in various areas. ANNs performed equally well with linear methods in 30% and better in 56% of the cases reviewed. In a subsequent study by Stock and Watson (1999) linear and nonlinear methods were compared and it was found that in terms of forecasting performance, combinations of nonlinear methods are better than combinations of linear methods. Additionally, feedforward neural networks (FNNs) that constitute a special ANN architecture performed equally well or better than traditional methods in more than half of the cases. Another study of the forecasting ability of different methods is the series of M-Competitions. The latest one, the M3-Competition (Makridakis & Hibon, 2000), examined the FNN topology and the most popular forecasting methodologies and commercial software in several test cases. The results showed that FNNs did not exhibit good performance, which is due to the nature of the available data. As pointed out by Balkin (2001) only 25% of the data sets exhibited strong nonlinearity, while the lengths of the series were insufficient for model building in most cases. Zhang (2003) pointed out that no single method is best in every situation and that combining different models is an effective and efficient way to improve forecasting accuracy, giving examples of previous work in hybrid methods that use neural networks. His paper proposed a hybrid FNN-ARIMA methodology where ARIMA is used to model the linear component and FNNs modeled the forecasting errors. The hybrid method outperforms the two component methodologies when they are used separately.

Another interesting question that arises in time series modeling is that among the plethora of candidates, only some of the variables should be selected for model use, as inclusion of all of them could result to a model of great size and low accuracy. One approach to deal with this issue is to employ particular types of multivariate analysis, such as the partial least squares (PLS) (Wold, Sjöström, & Eriksson, 2001) or the principle component analysis (PCA) (Hörnquist, Hertz, & Wahde, 2003) methodologies. However, PLS and PCA use latent variables and their major drawback is that the link back to the physical variables of the system is lost. Another common practice in marketing research is to employ statistical tests for variable selection (i.e. Poh, Yao, & Jašic, 1998). Combined statistical methodologies and artificial intelligence technologies, like fuzzy logic have also been applied for selecting the appropriate input variables in forecasting problems (Mastorocostas, Theocharis, & Petridis, 2001).

A variable selection method has to meet two conflicting objectives: The minimization of the modeling error and the minimization of the number of selected variables, which means that the optimization algorithm must compromise between the model's accuracy and parsimony. This can be achieved by combining multiple objectives into a single one. The final prediction error (FPE) (Akaike, 1969, 1970), Akaike's information criterion (AIC) (Akaike, 1974), and the modified FPE (Leontaritis & Billings, 1987) are standard selection criteria that have been employed for this purpose.

This paper is concerned with the development of time series sales forecasting models for short shelf-life food products, especially milk. Fresh milk has a shelf-life of less than a week and it is delivered and stocked on

shelves daily. Therefore, stockpiling by customers is unlikely to happen and it could only be limited to a small extent (Kondo & Kitagawa, 2000). Under these special circumstances it would certainly be beneficial for a milk production plant to be equipped with an accurate sales forecasting model combined with a flexible production system. An early example of such a system was developed by Schuermann and Kannan (1978) and its implementation would lead to a cost reduction of around 7%. Several studies that make use of milk sales data have been published concerning the effect of promotions on the sales of milk. Green and Park (1998) found differences in the reaction of customers to price and promotion changes in products that differ in fat content. Kondo and Kitagawa (2000) presented a methodology for time series analysis on milk sales that allows close examination of some factors that influence milk sales, such as trend, regular variation during weekdays and promotions. Kuo, Wu, and Wang (2002) proposed a method that integrates ANNs and fuzzy neural networks with fuzzy weight elimination and is able to model promotional activity, the effect of increased media exposure and the actions of the competitors.

In this work we apply a novel time series methodology to the problem of forecasting the daily sales volumes of short shelf-life food products, such as fresh milk. The methodology combines two artificial intelligence technologies: The radial basis function (RBF) neural network architecture for building the time series model and a specially designed genetic algorithm (GA) that selects the appropriate input variables to the model, based on the FPE criterion. The RBF neural network topology has a special structure that has certain advantages over the more popular FNN architecture, including faster training algorithms and more successful forecasting capabilities. GAs are mathematical methods that imitate the natural selection mechanisms and are mostly appropriate for solving discrete optimization problems. The combined GA-RBF method is applied on sales data of fresh milk provided by one of the leading manufacturers of dairy products in Greece. The results are compared to many standard time series methodologies, illustrating the efficiency of the proposed approach.

The rest of the paper is structured as follows: In the next two sections an overview is given on the artificial intelligence technologies that are employed in the GA-RBF time series method. In Section 4 the methodology is presented in details, while in Section 5 the results of the application of the method to true sales data are shown. The paper ends with the concluding remarks.

2. An overview of the RBF network architecture

RBF networks are in essence nonlinear modeling structures that unveil the mathematical relationships among the inputs $\mathbf{x}$ and the output y of a system ($\mathbf{x}$ is in bold indicating that it is a vector that may consist of more than one input variables). For the development of an RBF network only a training set of input–output examples $(\mathbf{x}_i, y_i)$, $i = 1, 2, \ldots, K$ is needed, that is a number of samples collected from the system. Further knowledge on the system is not required. RBF networks form a special neural network architecture that consists of three layers. The input layer is only used to connect the network to its environment. The hidden layer contains a number of nodes, which apply a nonlinear transformation to the input variables, using a radial basis function. The output layer is linear and serves as a summation unit. The typical structure of an RBF neural network with only one output node is depicted in Fig. 1.

Each hidden node is associated with a center, which is a vector $\mathbf{c}$ with dimension equal to the number of inputs to the node. The activity v of a hidden node is the Euclidean distance between the input vector and the node center. The activity is passed to the radial basis function and the response of this function is the hidden node output. In the present work, the thin-plate-spline radial basis function is employed:

$$f(v) = v \log(v) \tag{1}$$

A training algorithm aims at the determination of the structure and the parameters of the network, so that the error between the true and the predicted output values in the training set is minimized. In rigorous optimization notation, this minimization problem can be expressed as follows:

$$J(L, \mathbf{c}_j, w_j) = \sum_{i=1}^{K} (y_i - \hat{y}_i)^2 \tag{2}$$

where

$$\hat{y}_i = \sum_{j=1}^{L} w_j f\left(\left\|\mathbf{x}_i - \mathbf{c}_j\right\|\right), \quad i = 1, 2, \ldots, K \tag{3}$$

In the above equations, f is the radial basis function, L is the number of nodes in the hidden layer, $\mathbf{c}_j$ is the center of the jth hidden node, w_j is the connection weight

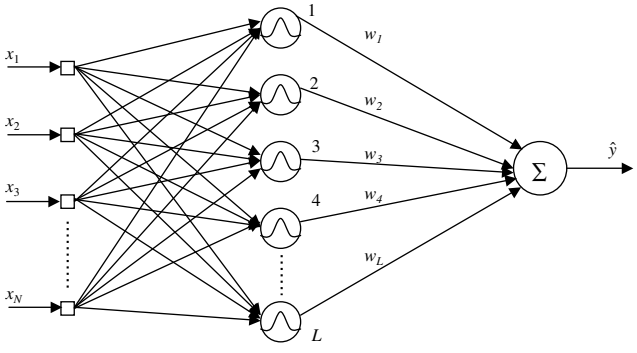

Fig. 1. The RBF neural network topology.

between the jth hidden node and the output node and $\|\cdot\|$ is the Euclidean norm. The set of equations defines a mixed integer nonlinear programming (MINLP) optimization problem, where the objective function must be minimized with respect to both the network structure and the network parameters (the hidden node center locations $\mathbf{c}_j$ and the output weights w_j, $j = 1, \ldots, L$). The problem cannot be solved by traditional optimization algorithms in a reasonable time. The training algorithms are specially designed methodologies that can approximate the solution to the aforementioned optimization problem as close as possible in a limited amount of time.

In this work we employ the fast and efficient fuzzy means clustering technique (Sarimveis, Alexandridis, Tsekouras, & Bafas, 2002). The fuzzy means algorithm is an innovative method, which has the ability to determine automatically the number of hidden layer nodes, while it proves to be orders of magnitude faster than the standard k-means algorithm (Moody & Darken, 1989). The method is described in details in the aforementioned reference.

3. An overview of genetic algorithms

GAs are machine learning procedures, which derive their behavior from the process of evolution in nature and are used to solve complicated optimization problems (Goldberg, 1989; Michalewicz, 1996). In nature, individuals that better fit the environment have more probability of surviving and transferring their chromosomes to their descendants, compared to individuals with poor fitness that most probably will become extinct. Following the same idea, GAs are iterative stochastic methodologies, that start with a random population of possible solutions (chromosomes). The fitness of each chromosome is measured by computing the corresponding value of a carefully chosen fitness function. Then, a new generation is produced by giving more probabilities of surviving to the individuals with the best fitness values. As the algorithm proceeds, the members of the population are gradually improved. This parallel searching mechanism is the main advantage of GAs, since they cannot easily get trapped in local minima. In order to better emulate the way nature behaves, some genetic "operators" are added to the algorithms, such as the mutation operator, where some members of each individual are altered randomly, and the crossover operator, where new individuals are born from a random combination of the old ones.

A genetic algorithm must involve the following components (Michalewicz, 1996):

- The representation of the possible solutions as chromosomes.
- A procedure for initializing the population of solutions.
- An evaluation function that computes the fitness of each individual.
- The genetic operators that are applied on the individuals to alter their characteristics.
- Values for the parameters used by the genetic algorithm (size of the population, probabilities of applying the different genetic operators, etc.).

4. The GA-RBF algorithm

The GA-RBF algorithm provides a complete framework for building time series models based on available data, since apart form providing a mathematical expression, it also selects the appropriate factors that are going to be used as inputs to the model. Therefore, it considers the problem as a small or medium scale variable selection problem (Kudo & Sklansky, 2000), since the candidates for becoming input variables in time series modeling are usually less than 50. The two objectives (selection of proper regressors and minimization of the prediction error) are combined into a single objective function. The methodology uses a specially designed GA as a search strategy, which employs a hybrid coding of genes. More precisely, each potential input variable is coded by a binary gene denoting whether the variable is present in the model (the gene has the value 1) or not (the gene has the value 0). An additional integer gene in each chromosome corresponds to the number of fuzzy sets that are defined in the domain of each variable, which is a parameter used by the fuzzy means algorithm. Thus, the length of each chromosome is equal to the number of candidates for becoming input variables plus one. Assuming that a set of input–output data $(\boldsymbol{x}_i, y_i)$, $i = 1, 2, \ldots, K$ is available, the GA-RBF methodology can be decomposed in the following steps:

Step 1: The population of P chromosomes is randomly initialized. Each chromosome consists of a binary string and an integer gene in the last position (representing the number of fuzzy sets in the fuzzy means training algorithm) that takes values between some predefined upper and lower limits $l_{\min}^{\mathrm{GA}}$, $l_{\max}^{\mathrm{GA}}$.

Step 2: The fitness value of each chromosome is calculated. The employed fitness function is based on the FPE criterion:

$$\mathrm{FPE}_m = \frac{2K + kn_m}{2K - kn_m}\mathrm{MSEP}_m, \quad m = 1, 2, \ldots, P \tag{4}$$

where n_m is the number of selected variables for the mth chromosome, MSEP_m is the mean squared prediction error of the RBF network corresponding to the mth chromosome measured on the available set of data and k is a statistical constant corresponding to the

significance level of the hypothesis testing criterion. For small scaled problems, the parameter k is set equal to zero. The RBF neural network model corresponding to a particular chromosome is constructed by using only the input variables that are represented by 1s in the binary genes of the chromosome. Then the fuzzy means algorithm is applied, where the number of fuzzy sets in each input direction is set equal to the content of the integer gene of the chromosome.

The fitness value for each chromosome is finally calculated as:

$$E_m = \frac{1}{\mathrm{FPE}_m}, \quad m = 1, 2, \ldots, P \tag{5}$$

since the GA is tailored to maximize the fitness function.

Step 3: A new population is generated by selecting individuals from the old population based on the previously calculated fitness of chromosomes. The reproduction is implemented as a linear search through a roulette wheel. Each chromosome is allocated a slot on the roulette, with size proportional to its fitness. A random number is generated and a copy of a chromosome passes to the mating pool only if the random number falls in the slot corresponding to the particular chromosome. This procedure is repeated P times in order to select Pchromosomes for the next generation. It is possible that some chromosomes may be selected more than once in accordance to the schema theorem (Michalewicz, 1996): the best chromosomes get more copies, the average stay even and the worst die off.

Step 4: The crossover and mutation genetic operators are applied on the new population. The crossover operator is employed to exchange genes between two chromosomes. For the specific methodology a one-point crossover scheme is utilized, where after some pairs of chromosomes are randomly selected based on the probability of crossover, they exchange strings of genes. During the crossover operation, the last integer gene is treated in the same manner as the binary genes and it is exchanged as well.

However, the different nature of the last gene necessitates a special treatment during the mutation operation. Thus, uniform flip bit mutation is applied to the binary genes with a probability equal to p_{um}, but nonuniform mutation with probability p_{num} is used if the gene selected for mutation is the integer gene that represents the number of fuzzy sets (Michalewicz, 1996). Nonuniform mutation is preferred over uniform mutation for altering the number of fuzzy sets, since it initially searches the space uniformly, but as the algorithm proceeds it performs a more local search.

Step 5: The algorithm returns to step 2, unless it has reached the maximum number of iterations or the fitness value has not been improved for the last L iterations.

The final outcome of the algorithm is the chromosome that produces the best fitness value (smallest value of the FPE criterion) during the entire procedure. This chromosome defines the optimal subset of input variables and the produced RBF time series model.

Remark 1. The fitness function (5) is formulated, so that it compromises between the predictive ability and parsimony of the model. That is, the evaluation function gives merit to models with minimal prediction errors, but also penalizes complex models that contain many variables.

Remark 2. Based on the above description the tuning parameters of the GA-RBF algorithm that must be defined by the user are the following:

- The size of the population P, which represents the total number of chromosomes.
- The total number of generations G.
- The number of consecutive generations for which the objective value is not improved M.
- The probability of crossover p_{c}.
- The probabilities of uniform mutation p_{um} and nonuniform mutation p_{num}.

5. Results

The problem under study is the evaluation of forecasting performance of the GA-RBF methodology on the daily sales of fresh full-fat milk in the area of Athens, Greece and more specifically, of the 1 l pack. Daily sales data of the 1 l pack for the first few months of the years 2001 and 2002 were provided by a leading manufacturer of dairy products. In order to incorporate the day-of-the-week effect and the effect national holidays have on sales, the data were analyzed and arranged, so that every day of the first year was appointed a corresponding day in the second year and thus, 108 pairs of corresponding days were formed. Distribution and sales at large outlets takes place 6 days a week, so there are data for 18 weeks. Some convenience stores or kiosks operate on Sundays, but they only account for a small amount of sales, which are absorbed in the sales of Saturday or Monday.

A question that arises at this point is how to utilize the data in order to exploit the information they contain. The data of the previous year are useful because they describe the actual course of sales in the corresponding week and provide insight into how sales are expected to behave. Past sales data from the current year contain the changes that have meanwhile occurred in the market and have affected the level and trend of sales. The change in trend could be fed into the model by providing it with the percentile change in sales between the current year and the previous year. Although it is possible to use virtually all of the data available till the day

before the one to be forecasted, it was decided that the most important candidate variables were the six previous days of the current year, the six previous days of last year, the percentile change between the two years as mentioned above and the corresponding day of the previous year, thus summing to a total of 14 candidate variables.

Current forecasting practice in the industry involves using some or all of the above variables and is often manual, that is rules of thumb are used instead of forecasting algorithms. Although it might seem that the best solution would be to formulate the model based on as much information as possible and therefore on all the available variables, this in practice often leads to the deterioration of the forecasting performance, due to possible correlations between the variables and duplication of information. On the other hand, too little information could leave the model with insufficient knowledge of the past.

The data of 108 days were divided into two sets of equal length: the first set (training dataset) was used for model building and parameter estimation and the second set (validation dataset) was used to test the model. The target was to build a model that can predict the 2002 sales volumes. The data of the year 2001 served only as historical information. This manipulation of data formulated a variable selection problem involving the 14 candidate variables mentioned before. The GA-RBF method was programmed in the Matlab 6.5 environment and was used to select the optimal subset of variables, employing the parameter values that are summarized in Table 1. The parameter k in Eq. (4) was set to 1. The execution time was less than 15 min in a Pentium IV 1400 MHz processor, which shows that the method is not computationally demanding, considering that the algorithm needs to run only once. The input variables chosen by the method were five: the sales of the current year with lags -1 and -6 and the sales of the previous year with lags -3, -5 and -6. This set of variables was also used to test the performance of various time series methods, which can be classified into linear, nonlinear and hybrid methods, as explained in the sequel.

Table 1
Parameters for the variable selection algorithm

Parameter	Value
Population size, P	20
Maximum number of generations, G	300
Maximum number of consecutive generations for which no improvement is observed, M	10
Crossover probability, p_c	0.75
Mutation probability, p_{um}	0.01
Nonuniform mutation probability, p_{num}	0.1
Lower bound of the number of fuzzy sets, l_{min}^{GA}	3
Upper bound of the number of fuzzy sets, l_{max}^{GA}	15

The multivariate ARMA procedure (Box et al., 1994) is in fact a combination of two models: the autoregressive (AR) part, where only past values of the time series data are used and the moving average (MA) model, where the forecast is generated from past values of the forecast errors. The two methods can be used separately or combined. The second linear method that is used in this work for comparison purposes is the Holt–Winters univariate procedure, which is a generalized exponential smoothing method that can incorporate trend and seasonal variation in the model (Chatfield, 1980). This method uses exponential weighting of the coefficients of past observations in order to give more weight to the most recent ones. We also tested hybrid methods for forecasting, that is combinations of linear and nonlinear methodologies, for example a linear AR model and a neural network MA model or vice versa. In all these methods, the RBF neural network architecture was employed and the fuzzy means algorithm was used as the training procedure. Finally we tested the incorporation of adaptation capabilities on the models that use only past values of the sales volumes as inputs (AR models). For the linear AR model, the recursive least squares (RLS) (Åstrom & Wittenmark, 1994) was utilized to modify the parameters over time in accordance to the changing patterns exhibited by the time series. The RBF neural network time series model was modified using the adaptive fuzzy means algorithm (Alexandridis, Sarimveis, & Bafas, 2003), that is able to correct on-line both the structure and the parameters of the model. Computational issues are not critical in all the aforementioned methods, since once the optimal subset of variables has been selected, execution times for all the methods are in the order of seconds.

The comparative results are summarized in Fig. 2, where for each time series methodology the mean absolute % error is shown calculated on the prediction dataset. We can observe that the five methods that scored poorest rely on linear models, which is an indication that the behavior of the time series should be classified as mostly nonlinear. It is useful to point out that the RBF model that used only past values of the sales volumes performed better than all the other configurations that involve neural network modeling. Not surprisingly, adaptation improved the predictions in both linear and nonlinear modeling, since it provides the models with up-to-date knowledge that is not available in the original training data. The improvement is more clear in the linear case. The results illustrate the efficiency of the GA-RBF method and indicate that the predictions can be further improved if the model is allowed to correct itself as new information becomes available. Fig. 3 shows the actual sales in the validation data set along with the forecasts of the static and the dynamic RBF time series models. For confidentiality reasons, the sales data are scaled between 0 and 1.

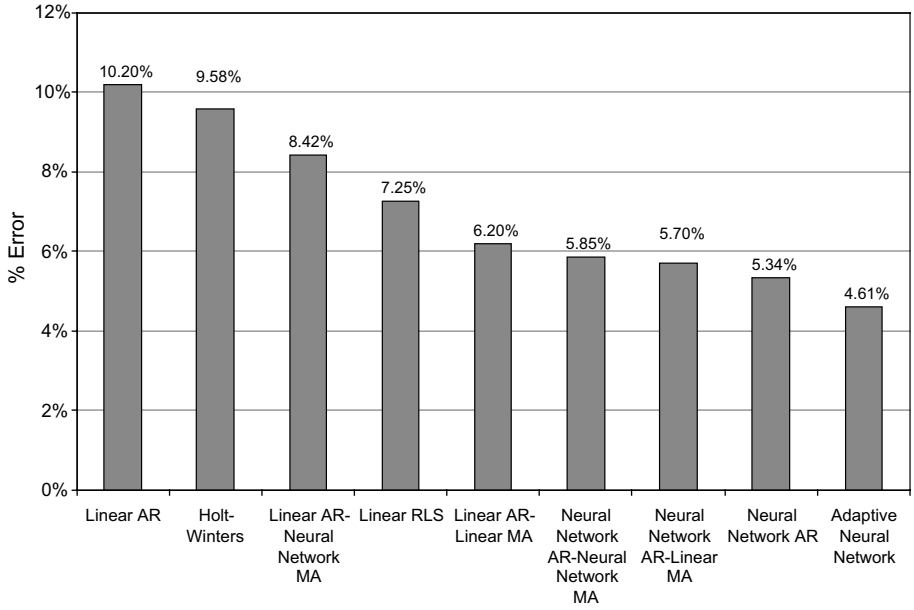

Fig. 2. Forecast error for different time series methodologies.

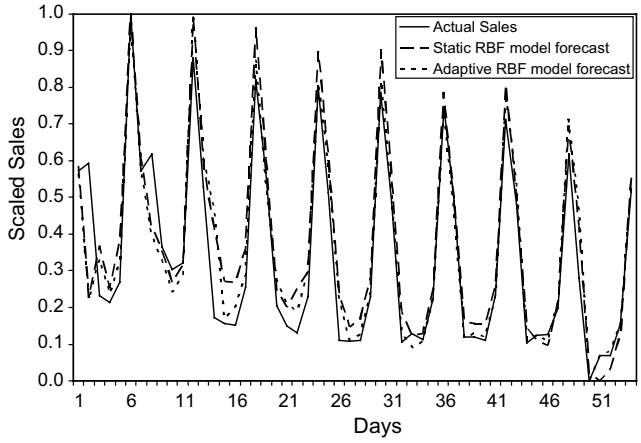

Fig. 3. Actual sales data and RBF model predictions.

The above discussion shows that an accurate time series sales model can be built for a particular product or a group of products using only historical sales information of the current and previous years. The type of model that is generated can range from a simple linear model that is produced using simple regression to an advanced adaptive neural network, where the accuracy is considerably improved for the price of an additional effort for developing the appropriate software tools. The models use only a subset of the available information (past sales data of both the current and the previous years), thus reducing the size and the number of model parameters. The selected variables are the most representative, among the much larger set of candidates. This fact does not mean that the other variables contain redundant information. There just exist a lot of correlations between the candidates, so that by selecting only some of them, most of the available information is retained.

The proposed method involves a number of tuning parameters that are summarized in Table 1. Experimentation with the values of those parameters showed that the method is not sensitive, when these parameters are fairly modified. Some guidelines on the selection of the tuning parameters are the following: The population size P, the maximum number of generations G and the number of generations M for which no improvement in the value of the objective function is observed, are in essence parameters that define the execution time of the GA-RBF algorithm. Depending on the available hardware and time, the user can freely increase the values of these three parameters. The most critical tuning parameters are the crossover and the mutation probabilities. The crossover probability is related to the crossover operator, which is the basis of the genetic algorithm. It encodes the probability that two selected organisms will actually breed and is typically selected between 0.6 and 1.0. The mutation operator is also important since it introduces some extra variability to the population and helps the training procedure to avoid getting trapped in local minima. However, high probabilities of mutation should be avoided, since this leads to almost random search. Recommended values are around the ones that are depicted in Table 1. Finally, on the lower and upper bounds on the number of fuzzy sets $l_{\min}^{GA}$, $l_{\max}^{GA}$ are tuning parameters that are used by the fuzzy means algorithm. The values that are depicted in Table 1 regarding these parameters are appropriate for addressing small to medium scale problems, which is usually the case for time series prediction problems.

6. Conclusions

In this paper we presented a complete framework for the development of sales forecasting time series models. The methodology is particularly useful for manufacturers of short shelf-life food products, such as fresh milk, since successful sales forecasting reduces considerably the lost sales and products returns. This is very important not only for the improvement of net income of the company, but also for environmental reasons since the returned milk is usually discarded. The GA-RBF method combines two advanced artificial intelligence technologies, namely the RBF neural network architecture and a specially designed genetic algorithm to select the appropriate explanatory variables and produce an accurate nonlinear time series model. Based on the application on true sales data, we illustrated the efficiency of the method and showed that the performance of the method can be further improved if we add adaptation capabilities to the neural network model, since this accounts for recent incidents that have not been considered in the original model. It should be pointed out that the GA-RBF method utilizes only historical sales data. In a future study we will show how additional information, like price, promotions, etc. can be explicitly taken into account in the development of the time series model.

Acknowledgments

Financial support by FAGE S.A. and the General Secretariat of Research and Technology in Greece under the PENED 2001 research program (01EΔ38) is gratefully acknowledged.

References

Adebanjo, D., & Mann, R. (2000). Identifying problems in forecasting consumer demand in the fast moving consumer goods sector. *Benchmarking: An International Journal, 7*(3), 223–230.

Agrawal, D., & Schorling, C. (1996). Market share forecasting: An empirical comparison of artificial neural networks and multinomial logit model. *Journal of Retailing, 72*(4), 383–407.

Ainscough, T. L., & Aronson, J. E. (1999). An empirical investigation and comparison of neural networks and regression for scanner data analysis. *Journal of Retailing and Consumer Services, 6*(4), 205–217.

Akaike, H. (1969). Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics, 21*, 243–247.

Akaike, H. (1970). Statistical predictor identification. *Annals of the Institute of Statistical Mathematics, 22*, 203–217.

Akaike, H. (1974). A new look at a statistical model identification. *IEEE Transactions on Automatic Control, 19*, 716–722.

Alexandridis, A., Sarimveis, H., & Bafas, G. (2003). A new algorithm for online structure and parameter adaptation of RBF networks. *Neural Networks, 16*(7), 1003–1017.

Åstrom, K. J., & Wittenmark, B. (1994). *Adaptive control* (2nd ed.). Addison-Wesley.

Balkin, S. (2001). The value of nonlinear models in the M3-Competition [Commentaries on the M3-Competition]. *International Journal of Forecasting, 17*, 545–546.

Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (1994). *Time series analysis: Forecasting and control* (3rd ed.). Englewood Cliffs, NJ: Prentice-Hall.

Chatfield, C. (1980). *The analysis of time series: An introduction* (2nd ed.). London: Chapman and Hall.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Reading, MA: Addison-Wesley.

Green, G. M., & Park, J. L. (1998). New insights into supermarket promotions via scanner data analysis: The case of milk. *Journal of Food Distribution Research*(November), 44–53.

Haykin, S. (1994). *Neural networks: A comprehensive foundation* (2nd ed.). New Jersey: Upper Saddle River.

Hörnquist, M., Hertz, J., & Wahde, M. (2003). Effective dimensionality for principal component analysis of time series expression data. *Biosystems, 71*(3), 311–317.

Kondo, F. N., & Kitagawa, G. (2000). Time series analysis of daily scanner sales: Extraction of trend, day-of-the-week effect and price promotion effect. *Marketing Intelligence & Planning, 18*(2), 53–66.

Kudo, M., & Sklansky, J. (2000). Comparison of algorithms that select features for pattern classifiers. *Pattern Recognition, 33*, 25–41.

Kuo, R. J., Wu, P., & Wang, C. P. (2002). An intelligent sales forecasting system through integration of artificial neural networks and fuzzy neural networks with fuzzy weight elimination. *Neural Networks, 15*(7), 909–925.

Leontaritis, I. J., & Billings, S. A. (1987). Model selection and validation methods for non-linear systems. *International Journal of Control, 45*, 311–341.

Makridakis, S., & Hibon, M. (2000). The M3-Competition: Results, conclusions and implications. *International Journal of Forecasting, 16*(4), 451–476.

Mastorocostas, P. A., Theocharis, J. B., & Petridis, V. S. (2001). A constrained orthogonal least-squares method for generating TSK fuzzy models: Application to short-term load forecasting. *Fuzzy Sets and Systems, 118*, 215–233.

Mentzer, J. T., & Bienstock, C. C. (1998). *Sales forecasting management: understanding the techniques, systems and management of the sales forecasting process*. Thousand Oaks, CA: Sage publications.

Michalewicz, Z. (1996). *Genetic algorithms + data structures = evolution programs* (3rd ed.). Berlin: Springer-Verlag.

Moody, J., & Darken, C. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation, 1*, 281–294.

Poh, H. L., Yao, J., & Jašic, T. (1998). Neural networks for the analysis and forecasting of advertising and promotion impact. *International Journal of Intelligent Systems in Accounting, Finance & Management, 7*, 253–258.

Sarimveis, H., Alexandridis, A., Tsekouras, G., & Bafas, G. (2002). A fast and efficient algorithm for training radial basis function neural networks based on a fuzzy partition of the input space. *Industrial and Engineering Chemistry Research, 41*, 751–759.

Schuermann, A. C., & Kannan, N. P. (1978). A production forecasting and planning system for dairy processing. *Computers & Industrial Engineering, 2*(3), 153–158.

Stock, J. H., & Watson, M. W. (1999). A comparison of linear and non-linear university models for forecasting economic time series. In R. F. Engle & H. White (Eds.), *Cointegration, causality, and forecasting: A Festschrift in honour of Clive W.J. Granger* (pp. 1–44). Oxford: Oxford University Press.

Van der Vorst, J. G. A. J., Beulens, A. J. M., De Wit, W., & Van Beek, P. (1998). Supply chain management in food chains: Improving

performance by reducing uncertainty. *International Transactions in Operational Research, 5*(6), 487–499.

Wold, S., Sjöström, M., & Eriksson, L. (2001). PLS-regression: A basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems, 58*(2), 109–130.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting, 14*(1), 35–62.

Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing, 50*, 159–175.