Neuro-Fuzzy prediction of airborne pollen concentrations

José Luis Aznarte M.[†], **Diego Nieto Lugilde**[‡],
José Manuel Benítez[†], **Concepción De Linares Fernández**[‡]
[†]Departamento de Ciencias de la Computación e Inteligencia Artificial
[‡]Departamento de Botánica
Universidad de Granada
jlaznarte@decsai.ugr.es

Abstract

In this work we applied several neuro-fuzzy models to the problem of airborne pollen forecasting. Experimental results show a great advantage of the neuro-fuzzy models against classical statistical methods, although there is still room for improvement. **Keywords:** pollen concentration, neuro-fuzzy, time series, forecasting.

1 Introduction

Forecasting future airborne pollen concentrations is undeniably of a high importance because of its medical, environmental and biological effects. The emission of pollen depends on a wide range of factors including temperature, humidity, wind etc. and so this is a highly chaotic and thus hard to model problem.

Application of classical methods to this problem has yielded results not entirely satisfactory. Models based on Soft Computing techniques have proved successful in a number of hard problems. So we decided to test different models from this area on this problem. In particular, we have selected different models which combine Fuzzy Systems and Neural Systems.

In this paper, we describe the application of neural and neuro-fuzzy models to the pollen concentration prediction problem and compare their performance with classical methods.

2 Pollen concentration data

The study was carried out on daily aerobiological data obtained over eleven years, from 1992 to 2003 inclusive, in the city of Granada (Southern Spain). Hence, around 4000 data points were available. Only *Olea Europaea* L. pollen values were considered. This species is one of the most allergenic in the Iberian Peninsula, and has a very strong seasonality. Choosing just one type of pollen produces a less noisy dataset, because the phenological behaviour is consistent. In addition, there are some statistical studies about this dataset, so more information is available for modelling.

To better model the time series some preprocessing of data is necessary. Besides of rescaling the dataset into the interval $[0, 1]$, special characteristics of the data suggests that further transformations could be used. In particular, high variance is normally tackled using a logarithmic transformation.

Notwithstanding, in this case, a linear log-like transformation was used instead, following the intervals used by the experts when dealing with this dataset (low, medium and high concentration, defined by the thresholds 50 and 200 grains/m^3). Fig. 1 shows this transformation. This transformation is oriented to the interpretation of the estimation errors on each of the intervals used by practitioners (i.e. for publishing forecasts on general media), and has no other justification.

Preprocessed data series was divided into two groups: dataset A comprised the years 1992 to 2002, both included, and was used for training/building the models; and dataset B, which

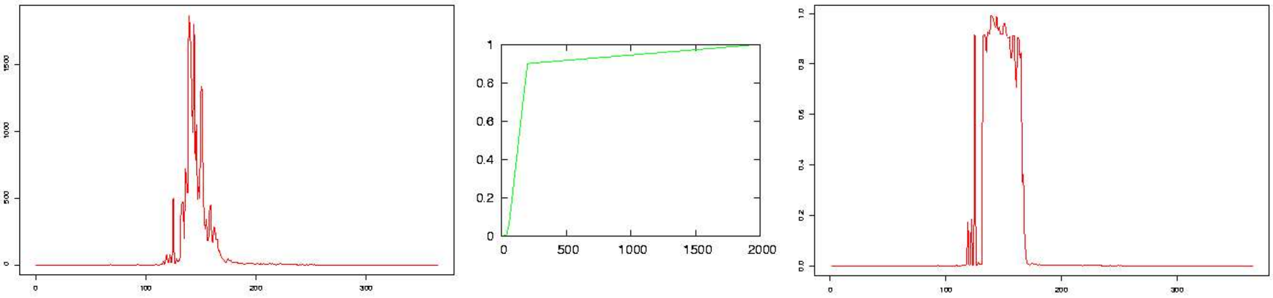

Figure 1: Sampled 2003 data, linear transformation applied to it and resulting 2003 data.

comprised only 2003 data, was used to test the models' performance.

To apply the Soft Computing models, we have to reshape the dataset into a compatible structure: a set of input-output vectors $[x_{t-1}, x_{t-2}, x_{t-3}, \cdots, x_{t-k}; x_t]$. The number of inputs for each point is obtained through the Box-Jenkins analysis. After this analysis, it was clear that the optimum $ARMA$ model was $ARMA(2,0)$ and so it had two AR components. This suggested that we should chose $k = 2$. Hence our vectors were of the form $[x_{t-1}, x_{t-2}; x_t]$.

3 Considered models

To test the applicability and performance of neural and neuro-fuzzy methods on this problem we have selected the following models: Multilayered Perceptron (MLP), Adaptive Neuro-Fuzzy Inference System (ANFIS), Hybrid Neuro-Fuzzy Inference System (HyFIS); Generalized Regression Neural Network (GRNN), and NEFPROX. To compare them, we have also selected two of the most popular classical statistical methods applied to time series modelling: AR-MA models and Holt-Winters Exponential Smoothing. In this section, a brief description of each model is provided.

3.1 Classical Statistical Models

3.1.1 AR-MA Models

The most popular class of linear time series models consists of autoregressive moving average (ARMA) models, including purely autoregressive models (AR) and purely moving-average (MA) models as special cases [1]. ARMA models are frequently used to model linear dynamic structures, to depict linear relationships among lagged variables, and to serve as vehicles for linear forecasting.

An *autoregressive model* of order $p \geq 1$ is defined as

$$X_t = b_1 X_{t-1} + \ldots + b_p X_{t-p} + \varepsilon_t \qquad (1)$$

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

Model (1) represents the current state X_t through its immediate p past values $X_{t-1}, \ldots, X_{t-p}$ in a linear regression form. It explicitly specifies the relationship between the current value and its past values. The model is easy to implement and therefore is arguably the most popular time series model in practice.

A *moving average process* of order $q \geq 1$ is defined as

$$X_t = \varepsilon_t + a_1 \varepsilon_{t-1} + \ldots + a_q \varepsilon_{t-q}, \qquad (2)$$

where $\{\varepsilon_t\} \sim N(0, \sigma^2)$, once again white noise. We note this model as $\mathrm{MA}(q)$, and it expresses a time series as a moving average of a white noise process. The correlation between X_t and X_{t-h} is due to the fact that they may depend on the same ε_{t-j}'s.

The AR and MA classes can be further enlarged to model more complicated dynamics of time series. Combining AR and MA forms together yields the popular autoregressive moving average (ARMA) model defined as

$$X_t = b_1 X_{t-1} + \ldots + b_p X_{t-p} + \varepsilon_t + a_1 \varepsilon_{t-1} + \ldots + a_q \varepsilon_{t-q}, \qquad (3)$$

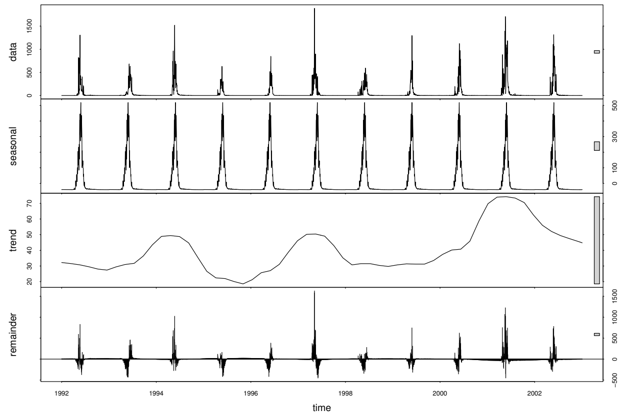

Figure 2: Decomposition of the series into seasonal and trend components.

where $\{\varepsilon_t\} \sim N(0, \sigma^2)$, $p, q \geq 0$ are integers and (p, q) is called the order of the model.

ARMA models are one of the most frequently used families of parametric models in time series analysis. This is due to their flexibility in approximating many stationary processes.

3.1.2 Holt-Winters Exponential Smoothing

The development of time series models begun with a modelling strategy called Classical Time Series Decomposition [8]. This approach consists of describing the behaviour of the time series through its non-observable components: Trend (T_t), Seasonality (S_t), Cycle (C_t) and Random Perturbation (ε_t). This is shown on Fig. 2. It is not possible to observe the trend of a time series, but it can be argued that it is linear, nonlinear or exponential. In the same way, it may be clear that in some specific periods of time, the values of the time series are higher or lower with a certain degree of regularity.

Exponential Smoothing techniques provide a way for predicting future time series values by weighting the influence of past observations. They are also sometimes called self-adaptive methods because once the parameters are estimated, the forecasts can be updated at each new observation. Holt-Winters method tries to express the time series as an additive or multiplicative combination of its components:

$$X_t = \mu_t + T_t + S_t + \varepsilon_{t-q}, \tag{4}$$

where μ_t is the exponentially weighted average of the past values of the series,

$$\mu_t = \alpha X_t + \alpha(1-\alpha)\mu_{t-1}, \tag{5}$$

and T_t, S_t are estimated from the data.

3.2 Multilayered Perceptron (MLP)

The MLP, trained by the standard backpropagation algorithm is the most widely used neural network approach for complex mappings forming between input and output, and its mathematical properties for non-linear function approximation are well-documented [2].

3.3 General Regression Neural Networks

The basic GRNN was published in 1991 by Donald F. Specht [7] as an extension of his Probabilistic Neural Network (PNN). It takes advantage of the concept that given a known joint continuous probability density function $f(\mathbf{x}, y)$ of a vector input $\mathbf{x}$ and a scalar output y, the expected value y given $\mathbf{X}$ can be computed by estimating the joint pdf using the Parzen estimator.

The core GRNN equation is

$$\hat{y}(\mathbf{x}) = \frac{\sum_{i=1}^{n} y_i \cdot h\left(\frac{\delta(\mathbf{x}, \mathbf{x}_i)}{\sigma}\right)}{\sum_{i=1}^{n} h\left(\frac{\delta(\mathbf{x}, \mathbf{x}_i)}{\sigma}\right)} \tag{6}$$

Specht designed the basic GRNN using Euclidean metric for δ and the Gaussian kernel for h.

3.4 Fuzzy Rule Based Systems for Time Series Analysis

The Fuzzy Rule Based System (FRBS) is a popular computing framework based on the concepts of Fuzzy Set Theory, fuzzy *IF-THEN* rules, and fuzzy reasoning [9, 5]. It has found successful applications in a wide variety of fields, such as automatic control, data classification, decision analysis, expert systems, robotics, pattern recognition and forecasting, to name a few.

3.4.1 Adaptive Neuro-Fuzzy Inference System

Adaptive Neuro-Fuzzy Inference Systems [3] are a special flavour of Fuzzy Rule Based Systems (FRBS) with an adaptation procedure which automatically tunes its parameters. It uses TSK-type rules of the form

$$\text{If } x_1 \text{ is } A \text{ and } x_2 \text{ is } B \text{ then } p(x_1, x_2) \quad (7)$$

which are trained by a hybrid learning algorithm. The antecedent parameters are trained by a gradient descent variant while the consequent linear parameters are trained by the least squares method.

3.4.2 Hybrid Neuro-Fuzzy Inference System (HyFIS)

As the ANFIS model described above, HyFIS [4] is a FRBS and a neural network. It employs Mamdani-type fuzzy rules of the form

$$\text{If } x_1 \text{ is } A \text{ and } x_2 \text{ is } B \text{ then } y \text{ is } C \quad (8)$$

which are tuned in a two-stage algorithm. The first stage deals with structure learning and fixes the number and configuration of fuzzy rules. The second stage, based on gradient descent as well, fine-tunes the parameters of the system to better model the training data supplied to it.

3.4.3 Neuro-Fuzzy Function Approximation (NEFPROX)

NEFPROX [6] is a Mamdani-type FRBS with a neural structure as the one used by ANFIS or HyFIS. It is closely related to NEFCLASS and NEFCON, and as well it has a two-stage learning algorithm, which fixes the structure of the FRBS on its first stage and then fine-tunes the parameters of the system via a heuristic procedure inspired in the gradient descent method.

4 Experiments and results

To test the performance of the models described in the previous section, we developed and carried out an empirical study. For this study we used the data described in section 2, transformed according to the piecewise linear function shown in Fig. 1.

Table 1: Experimental results using classical Time Series methods and Soft Computing methods over the transformed testing data.

Method	$RMSE$	Theil's U	# params.
naive	0.0918	0.3233	3
ARMA(2,0)	0.1029	0.3625	3
Holt Winters	0.1102	0.3881	3
MLP	0.0918	0.3234	
GRNN	0.0895	0.3153	1
HyFIS	0.0912	0.3211	
NEFPROX	0.0895	0.3296	32
ANFIS	0.0882	0.3381	45

The study was composed by two experiments. The first one aimed at establishing the overall performance of the selected models in one-step-ahead forecasts. Thus we built a model of each type using dataset A to tune its parameters. Then its forecasting accuracy was evaluated by measuring the Root Mean Squared Error (RMSE) of the predictions on dataset B. RMSE is defined as

$$RMSE = \sqrt{\frac{1}{N}\sum^{N}(\hat{y} - y)^2} \quad (9)$$

Aside from RMSE, we calculated the Theil's U statistic as well, in order to have a measure which is scale-free. Theil's U statistic is defined as

$$U = \sqrt{\frac{1/N\sum_i(y_i - \hat{y}_i)^2}{1/N\sum_i y_i^2}} \quad (10)$$

and it not bounded by zero and one. Large values indicate a poor forecasting performance.

The values of the parameters of the models were selected according to their corresponding authors' indications and after a little tuning through a short trial-and-error stage.

Results are summarized on Table 1. As can be easily seen the neural and neuro-fuzzy models yielded much better results than the classical approaches. Concretely, GRNN, NEFPROX and ANFIS produced the lower RMSEs, being the former the best. On the other hand, regarding the Theil's U statistic, the GRNN outperformed the other models, being again the Soft Computing approach more successful than the classical one in general.

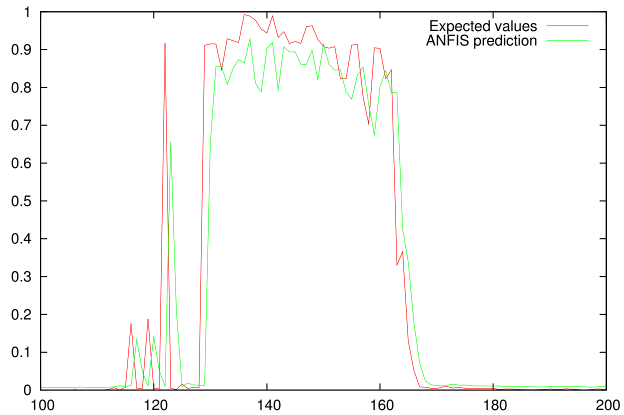

Figure 3: 2003 values and ANFIS' forecasts.

GRNN obtained better results in this case probably because it stores the whole training dataset and hence its forecasting abilities are better in tradeoff with its huge size.

Fig. 3 shows both the original 2003 data and the forecasts obtained by using the trained ANFIS. It is noticeable how the system manages to model the general behaviour of the data, including sudden rises and falls. Notwithstanding, a close look at this graph shows that the predicted values sometimes have a 1-2 days delay with respect to the original data. This may be explained by the fact that no exogenous variables (such as temperature, humidity etc) have been used, and the highly chaotic behaviour of the series is hard to model without that information.

The second experiment intended to show the accuracy of the one-step-ahead forecasts in terms of the prediction error, but this time we considered it segmented into the intervals that we used to transform the data. We wanted to know if the forecasts were equally good (or bad) on each interval, which could allow us to give the final user an estimation of the accuracy of the forecasts depending on the interval. Table 2 summarizes the RMSE results for the Soft Computing methods.

The lowest values of RMSE are obtained on the *low* interval for all models. This suggests that the models are able to model the data in this interval fairly good, which is not surprising if we consider that their values are all close to zero. The best model in this interval is ANFIS, followed closely by NEFPROX. The difference between models in this interval, though, is not high.

Table 2: Experimental results obtained on each transformed interval by applying the models to it.

Method	Interval	$RMSE_{test}$
MLP	*low*	0.0454
	middle	0.1712
	high	0.2665
GRNN	*low*	0.0289
	middle	0.1475
	high	0.2889
HyFIS	*low*	0.0373
	middle	0.1670
	high	0.2791
NEFPROX	*low*	0.0098
	middle	0.1484
	high	0.0458
ANFIS	*low*	0.0069
	middle	0.2142
	high	0.0484

The values obtained in the *medium* interval are not that good. Surprisingly, ANFIS obtains the worst result in this interval, being GRNN and NEFPROX the models which get the lowest value of RMSE. Again, there is not much difference between the results of all the models in this interval.

Finally, a few words about the error on the *high* interval. After the results of the models here, it is clear that the differences in overal error between the models come from their ability to predict values in this interval. As we can see in the table, ANFIS and NEFPROX are the only models that yield acceptable results on the *high* interval, while the others produce errors more than four times worse. MLP, GRNN and HyFIS models have big problems in predicting the peaks of the series. One cause for this problem may be that the 11 years that conform the training set show a big variation on the size of the maximum peaks, which rank from 400 to 2000 grains/m^3. This can be seen on Fig. 2. It is hence reasonable that the testing year, with over 1500 grains/m^3 on its highest peak is almost impossible to predict using this approach. The advanced training algorithms of NEFPROX and the type of fuzzy rules with linear consequents used by ANFIS, though, make these two models more appropriate to solve this task acceptably, and this justifies their better overall RMSE values.

5 Conclusions and final remarks

We considered the modelling of a highly chaotic, hard to model problem: the prediction of the olive tree airborne pollen counts in Granada (Southern Spain). The goal was to prove the applicability of neural and neuro-fuzzy techniques to solve this problem. We selected a number of approaches within this area and developed models to forecast the pollen concentration.

We also considered two classical approaches such as Box-Jenkins ARMA models and Holt-Winters exponential smoothing. The comparison between these models and the Soft Computing approach is clearly in favour of this one, although there is still room for improvements. Notwithstanding, the neuro-fuzzy approach has proved better for this problem, and this justifies further developments that would adapt neuro-fuzzy models to the specific task of Time Series Forecasting.

Acknowledgments

This research has been partially supported by Ministerio de Ciencia y Tecnología under projects TIC2003-04650 and TIN2004-07236.

References

[1] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control.* Holden–Day, 1970.

[2] G.E. Hinton D.E. Rumelhart and R.J. Williams. Learning representations by back-propagation errors. *Nature*, 323:533–536, 1986.

[3] J.-S.R. Jang. ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on systems, man and cibernetics*, 1993.

[4] J. Kim and N. Kasabov. HyFIS: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems. *Neural Networks*, 1999.

[5] G.J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic.* Prentice Hall, 1995.

[6] Detlef Nauck and Rudolf Kruse. Neuro-fuzzy systems for function approximation. *Fuzzy Sets Syst.*, 101(2):261–271, 1999.

[7] D.F. Specht. A general regression neural network. *IEEE Transactions on Neural Networks*, 1991.

[8] P.R. Winters. Forecasting sales by exponentially weighted moving averages. *Management Science*, 6:324–342, 1960.

[9] L.A. Zadeh. Fuzzy sets. *Information and control*, 3(8):338–353, 1965.