Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers & Operations Research 32 (2005) 2653–2670
computers & operations research
www.elsevier.com/locate/dsw

Approximating the sheep milk production curve through the use of artificial neural networks and genetic algorithms

Mercedes Torres, *, Cesar Hervás, Francisco Amador

Facultad de Ciencias Económicas y Empresariales, ETEA, Universidad de Córdoba, C/ Escritor Castilla Aguayo, no. 4 14004, Córdoba, Spain

Available online 25 January 2005

Abstract

This paper examines the potential of a neural network coupled with genetic algorithms to recognize the parameters that define the production curve of sheep milk, in which production is time-dependent, using solely the data registered in the animals' first controls. This enables the productive capacity of the animal to be identified more rapidly and leads to a faster selection process in determining the best producers. For this purpose we employ a network with a single hidden layer, using the property of "universal approximation". To find the number of nodes to be included in this layer, genetic and pruning algorithms are applied. Results thus obtained applying genetic and pruning algorithms are found to be better than other models which exclusively apply the classical learning algorithm Extended-Delta-Bar-Delta.

Keywords: Artificial neural networks; Genetic algorithms; Pruning algorithms; Non-linear regression; Extended-Delta-Bar-Delta; Gamma function

1. Introduction

Today there is an increasingly widespread use of artificial neural networks (ANN) as computational estimating models based on the adaptive learning process of a series of coefficients (weights) that represent the information that we intend to model. Although traditionally neural networks have been used to resolve problems related to pattern classification and recognition, there are more and more applications where the forecasting variable is of a quantitative nature (see [1,2]). The problem found in various application

* Corresponding author. Tel.: +34-957-222100; fax: +34-957-222101.
E-mail address: mtorres@etea.com (M. Torres).

0305-0548/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cor.2004.06.025

areas is that of establishing a functional relationship between the different variables forming part of the phenomenon under study. This problem has been traditionally approached using regression techniques to minimize a specific error function, prior to the researcher's choice of the model to be applied. In most cases, the model is non-linear which complicates the process. ANN have been recently used to resolve the type of problem which attempts to predict or estimate the value of a continuous function, f, which depends on various variables or characteristics $(x_1, x_2, \ldots, x_n)$ (see [3,4]). This is precisely the application employed in this study whose objective is to estimate the parameters defining the lactation curve in initial sheep production, thus enabling the productive capacity of the animal to be identified more rapidly. This leads to a faster selection process in determining the best producers through the use of quantitative criteria.

Among the different algorithms that have been used to train the network is the traditional Backpropagation algorithm (BP) first proposed by Werbos [5] and later used by Rumelhart and Mcdelland [6] which is a generalization of the delta rule [7], and, like this, undergoes a slow but sure learning process. Another algorithm employed is the Extended-Delta-Bar-Delta (EDBD) algorithm, later used by Williams and Minai [8] which is at the same time a modification of the Delta-Bar-Delta algorithm proposed by Jacobs [9] to improve learning speed.

One of the main problems involved in the application of ANN is the selection of the most appropriate network architecture to be used. That is, the correct number of nodes in the hidden layers of the network must be determined, as well as the amount of connections between the nodes in the different layers making up the network. Generally, the size of the network affects its complexity and the time necessary for training but, even more importantly, it affects its capacity to generalize (that is, its capacity to produce reliable and satisfactory results for data different than those used during training). In fact, the same learning error can be obtained in networks with different structures although the error of generalization would probably be different. In practice, it seems that a bigger network size contributes to a lower error level in the training set, although at the same time it could increase the error found in the generalization set. Because of this, there ought to be an analysis of the degree of complexity that an ANN should have to resolve each problem in order for the learning and generalization errors to be considered acceptable.

The first studies on the best possible architecture for an ANN tried to determine the optimal number of hidden layers. Hecht-Nielsen [10] formulated a theory based on Kolmogorov's theorem (1942) which affirmed that a network with a hidden layer could represent any arbitrary function of input. Later on, many authors, such as Carroll and Dickinson [11], Cybenko [12], Stinchcombe and White [13], showed that it actually was true that a network with a single hidden layer and activation functions, like a sigmoid, a hyperbolic tangent, etc., can approximate any continuous function. However, it has also been demonstrated that the degree of precision of these approximations can vary greatly, depending to a great extent on the number of nodes in the hidden layer. Some existing formal results show that the increase in the number of nodes in the hidden layer can reach exact approximations of continuous functions (see for example [14,15]), although this implies an increase in the complexity of the network. Our study is based on networks with a single hidden layer. To find the number of nodes to be included in this layer, as well as the number of connections between the nodes in the different layers that make up the network, the genetic algorithm (GA) proposed by Bebis and Georgipoulos [16] was used, coupled with Williams' pruning algorithm [17]. GA in the field of ANN have proved in many studies to be useful in the optimization of the network architecture and its weights (see [18,19]).

The GA are stochastic search algorithms that execute a global search in the weights' space, avoiding a fall to a local minimum that can often be produced by the overtraining of the network (see [20–22]).

In the case in question, GA is coupled with pruning algorithms to search for the neuronal network architecture that will allow us to verify, with a minimum of information, the parameters of the milk production function in the flock of sheep being analysed. So what we have here is an application of new computational methodologies for the management of a dairy, a sector relatively untouched by the technological innovations in business management that have been appearing in the last few years.

In general, the greater part of operational researchers' and agrarian economists' attention has been concentrated on the area of animal feed, due more to their connection with the animal feeding industry than to any connection with the dairy establishments themselves.

Among livestock activities, sheep farming has traditionally been forced into the background as far as researchers are concerned. Perhaps this is due to the limited interest afforded to the production of sheep milk on a worldwide scale in terms of quantity (which represents a mere 1.5% of total milk production and only 1.7% of total cattle milk production [23]). However, it must be kept in mind that the production of sheep milk occupies quite a relevant role since it constitutes the basis of a series of products of great nutritional value (like cheese and yoghurt, for example). At the same time, it is still one of the principle products of the very poor, also known as *subsistence* level, economies (the fact is that 55–60% of all sheep milk comes from countries with low or very low rates of income).

Among the main breeds of sheep found in Spanish flocks, the Manchegan breed is one of the most important. However, the production of milk obtained in the main area where this breed is found reaches only about 70 l per animal per lactation. This figure is very far from the real potential production of this race, since the total average production per animal on those livestock farms that carry out official milking controls is 166 l per lactation period, or 135 l in 120 days [24].

This study proposes models to identify the most productive animals in the flock. These models could lead to a decrease in the great differences in production that have been found in the last few years among different Spanish sheep breeds, like the Manchegan, with respect to other breeds (the French Lacaune, for example).

More precisely, the aim of this study is to predict the complete milk production curve of sheep, as determined by the coefficients A, b and c in the gamma function described below, using solely the data registered in the animals' first controls.

The early estimate of the curve has such different uses, such as: forecasting total animal production in cases of incomplete lactation; forecasting one animal's production, and consequently, the production margin that the sheep farmer can expect from his livestock; and, finally, providing a production forecast for lactation periods (120–150 days) in weaning programs [25]. To sum up, the objective is to predict the sheep's productive capacity and thus contribute to the elaboration of a selection program aimed at breed improvement.

In food science and in predictive microbiology [26], primary models present the currently accepted classification of prediction models, as they describe the change in the dependent variable over time and under given environmental and cultural conditions. These models can generate information about the form and values of the dependent variable such as lag-time, exponential growth rate and maximum population density, which are also known as kinetic parameters. Secondary models describe the response of one or more kinetic parameters, estimated from the primary model (e.g. lag-time), to changes in one or more of the environmental conditions (pH, temperature, additives, etc.). Tertiary models are applications of one or more secondary models to generate systems for providing predictions for non-modellers, with user-friendly software and Expert Systems. The issue dealt with here could be considered as a primary

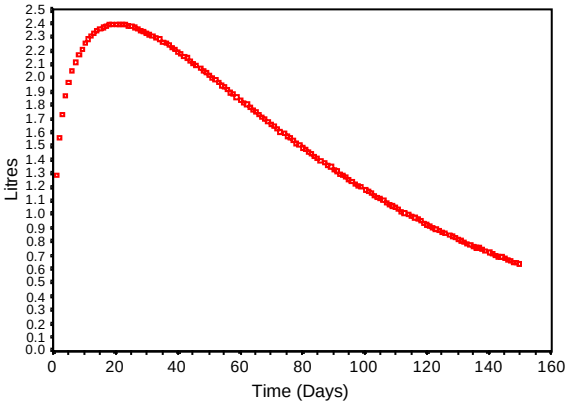

Fig. 1. Representation of the gamma function.

model, since it is one that estimates the lactation curve parameters using only production data from the first four lactation controls.

2. The lactation curve in sheep

The lactation curve in sheep flocks adjusts to the gamma function representing milk production in terms of time ($Y = At^b e^{ct}$). This function, represented in Fig. 1, was used initially by Wood in 1967 for cattle and seems to provide the best results for sheep according to the research of Treacher and Gribb [27] Carriedo and San Primitivo [28] and Molina [25]. As in other animal species, this curve presents, on the whole (see [29]), an ascending phase at the beginning of lactation until it reaches a maximum during the first weeks after birth, to later descend progressively until milk production dries up. Wood interpreted the parameters that define this function in the following way: *Y*—Daily amount of milk; *t*—time measured in days since birth; *A*—initial milk production. As the c parameter usually has values near zero; when $t = 1$, the milk production (*Y*) tends to *A*; *b*—slope of the curve in the ascendant phase; *c*—slope of the curve in the descending phase; *b/c*—this quotient coincides with the day of maximum production.

In the case of the Manchegan race, different gamma[1] functions have been proposed for each of the phases of milk production, resulting in the figures found in Table 1.

3. Methodology

ANN coupled with GA were applied to predict lactation curve parameters [30]. According to Munakata [31], an ANN is a computational model that attempts to simulate human behaviour in the learning process. In short, it can be affirmed that ANN is made up of a group of nodes or neurons connected to each other in a certain structure. It is precisely the composition of this structure (depending on the number of layers and

[1] To compare the value of the *A* parameter estimated by other researchers to that obtained in this study, one must consider the unit of measurement used for milk production because these previously mentioned estimates of milk production were expressed in milliliters/day and not in liters/day as in the case which now concerns us.

Table 1
Parameters of the gamma function for different phases of milk production (in ml/day)

Phase	A	b	c	R^2	Author and year
Lamb breeding	960	0.348	−0.027	0.980	Caja et al., 1992
Milking from birth on	795.6	0.230	−0.015	0.976	Gallego, 1983
Milking after weaning	0.009	3.723	−0.064	0.953	Fernández, 1985
	717	0.146	−0.014	0.994	Molina, 1987 [25]
	481	0.335	−0.018	0.980	Caja et al., 1992

Source: Gallego and Bernabeu [29].

neurons that it is composed of) that is one of the main problems to be considered when using ANN. Then, linear regression (LR) and quadratic regression (QR) were applied to predict lactation curve parameters and to compare the results with those obtained in the ANN models.

3.1. Learning algorithm

In order for a computational model of these characteristics to be called an ANN, it must be able to learn, that is, be able to recognize the value of the weights that represent the interconnections of the neurons or nodes found in the different layers making up the network. In this study, *supervised learning* is used to train and teach the network. In this method, a group of data called the *training set* is used to help the network to determine which values are appropriate for its weights. Each example is made up of an input signal with its corresponding correct answer or *target*. The learning process is as follows. An example from the training set is presented to the network, with its synaptic weights freely fixed at the beginning. If the desired response for the neuron j in the output layer of statement n is called $d_j(n)$ and the actual response obtained by the network was $y_j(n)$, then the error committed, $e_j(n)$, will be equal to the difference between the desired response and the real one.

Each input to the network, represented by $x_i(n)$ has a corresponding desired response, $d_j(n)$, and this pair of values constitutes a specific example which presents the network at the moment n. The learning process consists of modifying those weights that had been evaluated randomly at the beginning of the training to minimize the difference between the desired responses and those actually produced by the network. The training of the network is carried out for a considerable number of patterns until the network reaches the point where the weights no longer undergo significant changes. After carrying out numerous tests with the classical learning algorithm BP, the EDBD learning algorithm [8] was used since it offered better results. This algorithm introduces some modifications into the traditional back propagation (BP) algorithm to permit the elimination of some of the disadvantages attributed to the latter, such as: the slow processing of the convergence to a solution, the possibility of being trapped at a local minimum of the error function or even the possibility of not converging to the solution.

3.2. Determination of the best network structure: pruning algorithms and GA

The greatest difficulty involved in the use of ANN when solving any real problem is deciding the number of hidden layers that the network is going to have and the number of neurons to be included

in them. An inadequate choice could generate unsatisfactory results and could even induce the user to believe that the problem cannot be resolved with this methodology.

Based on the theory that ANN are universal approximators, and therefore an ANN with a single hidden layer can represent any arbitrary function of its input, networks with only a single hidden layer were trained.

With respect to the decision about how many neurons to include in the hidden layer, it was decided to start out with a network of a considerable size and to gradually eliminate the nodes and unnecessary weights until reaching a network size that provided satisfactory results. That is what is technically known as "pruning" a network. The Williams' pruning algorithm was the one used [17].

3.2.1. Williams' pruning algorithm

The pruning algorithm for regularization proposed by Williams comes from a network with a single hidden layer. The learning algorithm used is the BP algorithm. In each repetition, the weights selected as candidates for elimination are those which have reached a value equal to zero [16]. What has to be taken into account, however, is the fact that a weight can take a value equal to zero in a transition from a positive value to another negative one and should thus not be eliminated. To avoid this problem, the elimination of the weight will only be carried out if the null value reached is maintained throughout various repetitions, that is, when the derivative of the error function with respect to a specific weight is equal to zero. That would then indicate the existence of a minimum in the weights' space. Once this connection is eliminated, the weights' space is reduced by a dimension.

The algorithm aims to minimize a multiobjective cost function, represented in (1), composed by the addition of two terms. The first refers to the errors committed by the network in data estimation (that is, the difference between the real input and the desired target or output) the same as in other training algorithms; the second refers to the complexity of the network. So for two networks that reach the same estimation of data error, the cost function will penalize the more complex network to a greater degree. At the same time, this function allows for an equilibrium or balance in the generalizing capacity of the network and network structure. Thus, when faced with excessive pruning, the second term of the cost function will fall considerably at the cost of a steep increase in the first term, and vice-versa.

$$M(\vec{w}) = \beta\, E_{\mathrm{D}}(\vec{w}) + \alpha\, E_{\mathrm{w}}(\vec{w}), \tag{1}$$

where $E_{\mathrm{D}}(\vec{w})$is the deviation of the network in the data estimation, which is weighted for the coefficient β, which is greater than 0. This weighting factor represents the importance of the deviation. The second term, $E_{\mathrm{w}}(\vec{w})$, refers to network complexity, and is also weighted for the positive coefficient α, which is included to reflect the importance of network structure. (More details about this can be found in [32,33]).

3.2.2. Genetic algorithms

Genetic algorithms are inspired by Darwin's natural selection process. They are based on the collective learning of a population whose individuals represent potential solutions for the problem to be resolved. GA transfer a group of genetic individuals from one generation to the next. A set of individuals from the same generation are known as a population. Each population goes through a series of genetic operators of selection, recombination or variation (crossover or mutation) resulting in the next generation.

Among the different applications of the GA, we can find the search for the best neuronal network structure to resolve a concrete problem. The main objective of the GA is to find the individual with the greatest degree of aptitude or fitness. Each individual is a member of the population. The genotype of

an individual is made up of all the values of the weights in the network. The phenotype represents the present structure of the network and the values of its parameters.

GA is an attempt to create an intermediate population based on the fitness evaluation of each member in a current population. To evaluate the aptitude of each member, and since our intention is to determine the *structure* of the network that not only maximizes generalization but, at the same time, also minimizes the number of connections, we use an aptitude function optimizing these two characteristics. This function includes parameters that refer both to the size of the network and to its capacity for generalization. In (2) we can observe the multiobjective aptitude function that has been selected.

$$\text{Aptitude} = \lambda_{\text{GA}}^{(1-\text{net size})} + \lambda_{\text{SEP}}\, \text{e}^{-5\ \text{average SEP}}. \tag{2}$$

The parameters included in the aptitude function are explained below.

λ_{GA} (1-net size): this term is introduced in order to relate the size of the network with its own generalizing capacity. The λ_{GA} factor weighs the relationship of the network size with its ability to generalize. This factor is responsible for not attaching too much importance to network size until an appropriate degree of generalization is achieved. This factor is defined in (3), where λ_{0_GA} and $\beta_{\text{GA}E_{\text{gen}}}$ constants are defined by the user.

$$\lambda_{\text{GA}} = \lambda_{0\_\text{GA}}\, \text{e}^{(-\beta\text{GA}\ E_{\text{gen}})}. \tag{3}$$

The term *net size* refers to the relative network size if we were to prune at this very moment. It is calculated by the quotient between the number of weights that would make up the network after the pruning, represented by W_j, and the total number of weights before the pruning, W_i

$$\text{net size} = \frac{W_j}{W_i}. \tag{4}$$

$\lambda_{\text{SEP}}\, \text{e}^{-5\ \text{average SEP}}$: this term is included to eliminate the possibility of finding two individuals in one population presenting the same aptitude due to having the same network size and the same generalization error (E_{gen}). This is more probable if there are only a few validation patterns. This is why a function is introduced since two individuals would rarely coincide in the same E_{gen}. In this way, we have used the standard error of prediction (SEP), a relative error that is expressed in percentage. The coefficient is a non-dimensional variation associated with relative error. The SEP is the quotient between the typical error deviation committed by the network in the estimation of the objective parameter, and the average value of this parameter in the generalization set, as can be observed in

$$\text{SEP}_j = \frac{1}{\bar{t}_j}\sqrt{\frac{\sum_{K=1}^{N}\left(o_{jk} - t_{jk}\right)^2}{N}}; \quad \forall j = i, \ldots, \text{output}; \quad \forall k = 1, \ldots, N, \tag{5}$$

where o_{jk} is the value estimated by the network for the j exit in the k pattern, and t_{jk} is the target value for the j exit target in the k pattern and N is the number of patterns in the generalization set.

The term λ_{SEP} is a weighting factor fixed at the beginning of the GA execution. This value can be chosen in the range [0,1] by the user or automatically by the program. This value is fixed depending on if it is used to solve a problem of classification or of recognition. As the problem at hand is one of recognition, the value of 1 is assigned because the SEP coefficient is the most appropriate measurement for cases of recognition.

The term $e^{-5\,\text{average SEP}}$ represents a function of an average SEP which is a weighted average of the SEP's found at each ANN output, as shown in (6). This operation permits some network outputs to be assigned a greater priority to optimize them. Of course it would also be possible to give the same importance to any network output.

$$\text{average SEP} = \sum_{j=11}^{k} \frac{|\text{SEP}_j| W_{\text{SEP}_j}}{k}, \tag{6}$$

where j is each network output, k the total number of outputs and W_{SEP_j} is the weighting factor which represents the relevance of the j output SEP coefficient, the SEP will have a value between 0 and 1, although higher values could be possible. To concede a higher aptitude for the individuals with average SEP's over 0 and less for those whose SEP's are closer to 1, the exponential function $e^{-5\,\text{average SEP}}$ was established empirically.

In the selection process, each member of the population produces a number of copies that is proportional to its fitness. This algorithm uses the roulette method [34] to select the individuals that form part of the new population, although a variation is included. The roulette method could cause the premature convergence of the solution. This would happen if the members with the highest evaluation put too much selective pressure on the new population, since they represent a high percentage of the population, thus eliminating diversity among its members. Because of this, there is a process of change in linear scale to control the number of copies that members with a high degree of aptitude would receive in future generations. The linear scale described in (7) is used with great frequency (f'representing the rise in aptitude and f the original aptitude). The a and b coefficients are calculated in each generation so that the maximum value of the aptitude in the new scale results in a small number.

$$f' = af + b. \tag{7}$$

GA uses an *elitist* strategy that guarantees that individuals with greater aptitude in a population will form part of the next generation. This is achieved by ordering the members of a population according to their aptitude and selecting the best for their survival in the next generation. The number of individuals that pass "intact" to the new generation depend on the parameter denominated G_{ap} which determines the percentage of the population that will be replaced in each generation. In this way, if N_{p} is the size of the population, the number of individuals that will remain intact in the following generation, represented by I, *will* be given by as

$$I = N_{\text{p}}(1 - G_{\text{ap}}). \tag{8}$$

Once the best members of a new population have been copied, the rest will be selected by the roulette method, with scaled aptitudes. After the reproduction has been carried out, the crossover operator will be applied, that is to say, the interchange of nodes and connections will take place among individuals selected randomly from an intermediate population to form new members. The GA used proposes a modification of the traditional crossover operation. The cross should only take place among the weights entering the first hidden layer in the network, because the function carried out by these is different than that of the weights in deeper layers. Thus the weights in the first layer act as feature detectors, while those weights in intermediate layers are indicative of acquired knowledge. According to this theory, it makes no sense to interchange weights in hidden layers since each network will represent knowledge in different ways. Therefore the weights entering the hidden layer will recombine according to a determined probability.

The last operator used by this algorithm is that of *mutation*. This operator randomly selects a member of the population and changes a set of its weights. What should be previously established is what probability exists of carrying the mutation through to fruition. In this work the mutation only takes place among the weights related to one and the same node or neuron, through the addition of a slightly different random change for each weight concerned.

The process of GA application comes to an end when the average improvement in the aptitude reached at a specific stage, represented by n, is lower than the average improvement found in the previous stage, $n-1$. To measure this improvement, Eq. (9) is used. Here A_n is the best measurement in the n stage; g is a constant whose value is selected randomly, normally close to the unit, and I_n is the best one found in the nth generation (namely by n) and defined as the quotient between the average of the aptitudes shown by the individuals in the nth generation, n, and the average of the aptitudes manifested by the individuals in the previous generation, $n-1$.

$$A_n = g\,A_{n-1} + (1-g)\,I_n. \tag{9}$$

3.3. Parameters of the evaluation of the findings

The evaluation of the results obtained from each ANN model will be carried out with respect to its generalization capacity and its size, because the objective is to find network structures that possess a good degree of generalization capacity with the least number of connections possible.

The generalization capacity will measure the ability of the network to provide correct results for different patterns than those used in the training set, and, where appropriate, than the validation one. To measure the ability for generalization, the generalization data set is used and we used the SEP coefficient (see Eq. (5)).

Logically, the higher the SEP coefficient value, the lower the degree of recognition by the network.

The *size of the network* indicates the number of connections that have not been pruned. It is a question of determining the minimum number of connections needed by the network to obtain a certain capacity of generalization. If the EDBD algorithm were to be applied unpruned, the network would maintain all its connections.

4. Practical application

The practical application of this study is based on the genealogical and productive records pertaining to a 21-year period (1980–2001) on a sheep farm that bred the Manchegan race of sheep that was located in the Spanish province of Ciudad Real. Although there were changes in the size of the flock during this time, there are currently still 3000 mothers in the production phase. Nonetheless, the lack of a systematic and rigorous method for the gathering of information (traditionally carried out by the shepherds themselves) made the filtering of the initial data a long and arduous task. Due to this, only those registers of original data with the most complete information were selected, thereby limiting considerably the useful data at our disposal. The lack of efficiency in the cataloguing of sheep flock data is not of particular importance in this study, however, although it is a frequent problem found by researchers in the sector and considerably limits progress in the current selection process of the Manchegan breed [24].

4.1. Data

Production data collection was carried out in the following way: once the sheep had given birth, the first control was registered on the day of the week specified for this task, before the eighth day after birth, to give the controls a week-long time span. The suckling period of the lambs lasted between 35 and 50 days, according to the growth reached and taking into account whether there had been a single or double birth. During this period, the lambs were separated from their mothers for 12 h before the control (only one daily control was taken during this phase, multiplying by two the production obtained in the first milking.) Once the lambs had been weaned, the sheep were milked twice daily until the end of the lactation period. The production was measured by volume, taken in two daily milkings after weaning and always without revision.

In the flock analysed, all the sheep were milked from the moment of birth (which took place between 1990 and 1997). In all the cases used to make up the curve estimation, the lactation period lasted 90 days or more. What must be emphasized here is the enormous difficulty posed by the extraction of data in these circumstances, since it is necessary for the shepherd himself to record the data out in the country.

4.2. Structure of the ANN employed

Feed forward neuronal networks with an input layer, an output layer and a hidden layer were used in the application of ANN. The input variables for this net were the average daily milk production figures corresponding to the second to fifth fortnights of lactation (the production after the first 2 week period was not used because this data was not recorded in many cases). These variables are normalized in the range (0.1, 0.9) to avoid the sign saturation problems found in the sigmoid transfer functions that are used in ANN models. Furthermore, this type of model does not take into account the type of correlation that could exist between the input variables, which is why they are specifically proposed in these cases. The need for employing production data averages for 15-day periods originated from the lack of milking controls in many cases during different weeks of the lactation period. This obliged a modification to be made in the amount of data included, and the application of the estimation of curve only in those cases that provided sufficient information, leaving a final count of 80 cases in the first lactation, 83 in the second and 72 in the third.

The exit-layer variable was to be the value of parameters A, b or c corresponding to the lactation curves of each animal analysed in the flock. Therefore, first of all, in order to prepare the lactation curve, non-linear regression (NLR) was applied to estimate the parameters of the gamma production function. The analysis was carried out with ANN of the A, b and c parameters of the gamma function previously estimated with NLR. Production data from 80 sheep in the first lactation were used, divided randomly between the training and generalization sets, leaving 55 patterns for training (68.75%) and 25 for generalization (31.25%). Second and third lactation production data (83 and 72 cases) were distributed in the training and generalization sets in the same proportions established in the first lactation analysis. Secondly, ANN was used to determine these parameters with the least possible information. Thus the estimated values of the A, b and c parameters in the production curves with NLR would be the values to be used as the point of departure or target in the next ANN training session.

In the estimate of the curve parameter with NLR, the dependent variable represented the production registered in each of the milking controls while the independent variable was represented by time, measured by the number of days from birth to each of the above-mentioned controls.

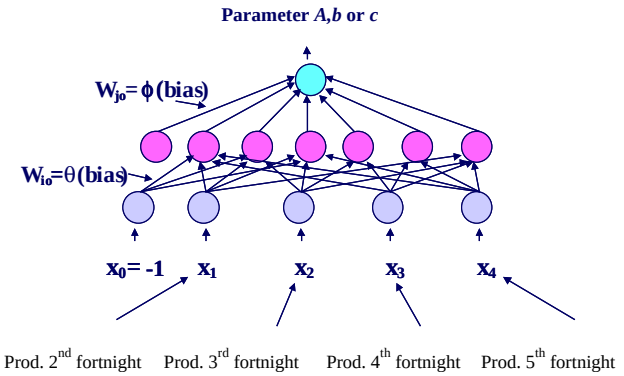

Fig. 2. Graphic representation of the ANN used to recognize production function parameters.

The number of nodes in the hidden layer was selected experimentally, beginning with models with a large enough number of connections a priori to be able to eliminate the insignificant ones. The maximum number considered was 10, which is large enough if we remember the number of input variables (only 4). The minimum number of nodes in the hidden layer was 4, one for each input variable. The other number of nodes considered (6) was selected randomly between the minimum and the maximum. Our experimental studies showed that six-nodes in the hidden layer gave the best results because a higher number caused over-training, that is, while the SEP values for the training set were small, those for the generalization set were greater. Hence the net architecture used was 4:6:1, as can be duly found in Fig. 2.

An initial application of the learning algorithm EDBD was carried out and, afterwards, the genetic and pruning algorithm already described were applied in order to compare the results obtained with different algorithms.

5. Results

5.1. Results obtained with NLR

In Table 2, some descriptive statistics are shown for each lactation on production curve parameters estimated with NLR. In this case the complete productive information of each sheep was used for the estimation. The estimated values with this information would later become the target values used in the ANN models during the training and generalization process.

As can be deduced from the above table, the problem of estimating the parameters of a lactation curve can be complex because the parameters that should recognize the ANN are highly dispersed, especially parameters *A* and *b* whose Variation Coefficients are nearly 100%.

The average value of the *R*-squared coefficient (R^2) fluctuates around 80%, with a variation coefficient of about 17% in the three lactations analysed. The percentage of production variability explained by the model was less than that found by other researchers (about 98%).

Table 2
Descriptive statistics on lactation curve parameters estimated with NLR

Parameter	Minimum	Maximum	Average	Typical deviation	Variation coefficient
First lactation					
A	0.00094	2.02280	0.4690	0.49220	104.95
b	0.05870	2.54829	0.76618	0.60640	79.14
c	−0.05995	−0.00870	−0.02221	0.01171	52.72
R^2	0.57261	0.95829	0.78240	0.10110	12.92
Second lactation					
A	0.00088	2.18817	0.58552	0.56859	97.11
b	0.01229	2.39110	0.68270	0.55256	81.38
c	−0.05430	−0.00561	−0.02100	0.01010	48.09
R^2	0.37144	0.96876	0.77230	0.13495	17.47
Third lactation					
A	0.00420	2.44400	0.62724	0.60939	97.15
b	0.00335	2.13030	0.69767	0.53154	76.19
c	−0.04920	−0.00582	−0.02200	0.01050	47.73
R^2	0.17030	0.97911	0.81254	0.14067	17.31

5.2. Results obtained with ANN

To facilitate the comparison of the results found with EDBD and with GA in the three lactations analysed, Table 3 registers the average SEP coefficients found in the generalization set in the 6 measurements performed with each algorithm to calculate each parameter of the lactation curve. It can be seen that we also used the mean squared error of prediction (MSE) for each parameter. In the case of the GA and pruning algorithm applications, the average number of connections has also been added as an indicator of the network size, which in turn gives us an idea of the network's complexity. In the case of the application with EDBD, the network would end up completely connected, thereby having 31 connections. Table 3 highlights the best results to facilitate their identification.

It can be deduced from the results shown in this table that the SEP variation coefficients have high values in the case of the A and b parameters, although in this case we must consider the great variability presented by the original data and thus, the scant representative capacity of the average values of these parameters in the generalization group (the Pearson variation coefficients of the A and b parameters in the generalization group were between 75% and 90%). It is precisely this great variability that makes the problem of determining the parameters for the ANN so much more complex. In the case of parameter c, whose distribution of values was much more homogeneous, the SEP coefficient showed much smaller values when GA was used.

As far as size network is concerned, we can observe how the use of GA and pruning meant a 30–60% decrease in the number of connections found with respect to the size of the completely connected net. Thus, with this method of architectural design in the net models, which use genetic algorithms with real codification as well as connection-eliminating algorithms with insignificant weights, the majority of the models present an average of between 10 and 20 weights to estimate. With this reduced

Table 3
Average SEP, connection number (CN) and MSE, found in 6 executions of the algorithm with different proposed network models

Lact.	Parameter *A*						Parameter *b*						Parameter *c*					
	EDBD			GA			EDBD			GA			EDBD			GA		
	SEP	CN	MSE	SEP	CN	MSE	SEP	CN	MSE	SEP	CN	MSE	SEP	CN	MSE	SEP	CN	MSE
1	48.3	31	0.05	54.1	20.5	0.06	48.6	31	0.19	48.9	16.2	0.19	50.5	31	1E-04	25.2	18.8	3E-05
2	55.9	31	0.06	49.7	19.5	0.05	42.0	31	0.08	40.0	22.5	0.07	29.5	31	3E-05	17.7	19.2	1E-05
3	56.9	31	0.21	52.8	13.2	0.18	45.4	31	0.09	63.2	19.2	0.17	55.7	31	1E-04	17.3	12.0	1E-05

number of connections the model proposed preserved a good generalization capacity for each of the lactation curve parameters and each of the 3 generalization sets used. This result provides a high degree of interpretability for the proposed models which is fundamental in changing the idea that ANN are "black-box" models.

5.3. Results obtained with LR and QR

As well as the ANN, other statistical techniques were applied to estimate the parameters that define the lactation curve, with the aim of identifying the methodology that offers the best results. Linear regression was applied to predict each of the parameters that define the lactation curve, in each of the three lactations analysed. In the linear regression model the depending variable was made up of the value of the parameter (*A*, *b* or *c*, according to each case) which previously had been estimated by NLR with all the production information from each sheep in the set of data used for training in the application of ANN. So, the dependent variable on the regression model was the output variable in the application of ANN. The productions registered in the second to fifth lactations were used as independent variables (those that were input variables in the ANN application). Thus, for example, for the estimation of parameter *A*, the linear regression model to be estimated in each lactation would be the one represented in (10), with P_2, P_3, P_4 and P_5 being the productions registered in each of the 4 fortnights of lactation considered. (Regression trough the origin was considered because the intercept was not significant).

$$\hat{A} = \beta_1 P_2 + \beta_2 P_3 + \beta_3 P_4 + \beta_4 P_5. \tag{10}$$

Given that significant correlations were found among the independent variables, as each of them represented milk production at different moments in time, certain transformations of the original variables were carried out (new variables were calculated from the difference, the quotient and the increase in production between two successive fortnights). However, while the multi-colinearity problem was solved in this way, as the tolerance statistics increased considerably, auto-correlation of errors was generated and the model's capacity for prediction was weakened (the co-efficient of linear determination lessened). Multi-colinearity makes the interpretation of regression coefficients difficult, but this is not a serious problem when the sole purpose of the regression analysis is prognosis or prediction, as the higher the determination coefficient (R^2), the more accurate the prediction [35].

As the final aim of this study is not to learn the individual contribution to the value of the curve parameter for each fortnightly production, but to compare the predictive capacity of ANN with other traditional techniques for statistical estimation, in the end regression models were applied to the original variables, without transforming them, as these are the ones that lend themselves to a better level of adjustment both in the training set and in the generalization set. The variables whose coefficients were short of a significance level of 5% were eliminated, as it was seen that the predictive capacity of the model did not deteriorate but even improved slightly, as the level of adjustment in the general group increased if the non-significant regression coefficients were previously eliminated from the regression model, (in spite of the fact that multi-colinearity produces wider confidence intervals for regression coefficients, and means the more ready acceptance of the null hypothesis that the population regression coefficient is zero).

The adjustment of quadratic regression models was also carried out, which take into account the interaction between independent variables. So, for example, the model of quadratic regression to estimate

Table 4
MSE and SEP obtained in the generalization set with the regression models considered

Lactation	Parameter	Linear model	MSE	SEP	Quadratic model	MSE	SEP
1	A	$0.704P_2 - 0.333P_4$	0.19	97.06	$0.704P_2 - 0.333P_4$	0.19	97.06
2		$1.057P_2 - 0.767P_5$	0.08	63.20	$-1.62P_4 + 2.14P_5 + 1.26P_2^2$ $-0.83P_3^2 - 1.85P_2P_5 + 1.27_3P_4$	0.14	83.94
3		$0.812P_2 - 0.474P_4$	0.40	78.27	$0.97P_2 - 0.36P_5 - 0.19P_2P_4$	0.45	82.65
1	b	$0.495P_4$	0.61	87.16	$-1.76P_2 + 3.23P_5 + 0.48P_2^2 - 1.11P_5^2$	0.38	68.49
2		$1.216P_4 - 0.728P_2$	0.16	59.77	$-1.03P_2 + 4.69P_4 - 2.78P_5 + 1.04P_3^2$ $-3.25P_4^2 - 2.31P_3P5 + 4.35P_4P_5$	0.23	71.73
3		$0.558P_5$	0.45	102.88	$-0.53P_2 + 1.33P_4 + 0.06P_5 - 0.22P_4^2$	0.28	81.43
1	c	$-0.015P_4$	2.29E-04	62.74	$-0.002P_2 - 0.03P_5 + 0.01P_5^2$	1.6E-04	52.49
2		$-0.014P_4$	4.07E-05	31.89	$-0.02P_4 + 0.003P_2^2$	3.9E-05	31.50
3		$-0.014P_4$	1.97E-04	66.67	$-0.01P_2 - 0.02P4 + 0.008P_2P_5$	1.4E-04	57.93

for parameter A is

$$\hat{A}=\beta_1P_2+\beta_2P_3+\beta_3P_4+\beta_4P_5+\beta_5P_2^2+\beta_6P_3^2+\beta_7P_4^2+\beta_8P_5^2 + \beta_9P_2P_3+\beta_{10}P_2P_4+\beta_{11}P_2P_5+\beta_{12}P_3P_4+\beta_{13}P_3P_5+\beta_{14}P_4P_5. \quad (11)$$

The Levenberg–Marquardt method was applied, and regression coefficients were eliminated if they were short of the 5% significance level.

The regression model obtained in each case for the training set was used to estimate the parameter in the generalization set. The SEP coefficient was estimated for each parameter and lactation in the generalization set. Table 4 shows the results (MSE and SEP) obtained in the estimation of parameters with linear and quadratic regression models in the generalization set.

As can be seen in Table 4, with the linear regression models, better estimations were obtained for parameter A than with the quadratic models. However, in the case of parameters b and c, the quadratic models were better. Nevertheless, we can state that the SEP coefficients were higher in all cases than those obtained with ANN, so the estimations with regression models were not accurate.

As a final step, the results obtained with ANN and Regression Models were compared. Table 5 shows the value of the SEP coefficients obtained with each model and lactation. As we can deduce from the results reflected in Table 5, the ANN gave better results than the regression models considered. The results obtained with GA were better than those produced by the learning algorithm EDBD in 66% of the cases. Although the SEP_G coefficients of parameters A and b are high, we can declare the superiority of the ANN with respect to the linear and quadratic regression.

The results obtained with GA were higher than those provided by EDBD in the majority of cases, especially in the determination of parameter c.

Table 5
SEP obtained with ANN and Regression Models

Lactation	Models	Parameter *A*	Parameter *b*	Parameter *c*
1	ANN (EDBD)	48.31	48.58	50.55
	ANN (GA)	54.09	48.91	25.24
	LR	97.06	87.16	62.74
	QR	48.31	68.49	52.49
2	ANN (EDBD)	55.96	42.03	29.52
	ANN (GA)	49.73	40.00	17.67
	LR	63.20	59.77	31.89
	QR	83.94	71.73	31.5
3	ANN (EDBD)	56.93	45.45	55.72
	ANN (GA)	52.78	63.22	17.34
	LR	78.27	102.88	66.67
	QR	82.65	81.43	57.93

6. Conclusions and future work

The specification of the lactation curve, through the use of productive information from the first controls made, can be used in incomplete lactations to estimate a part that is missing. This then constitutes the extended lactation when combined with the part known [36], permitting the recuperation of information that would otherwise be lost, and allowing a real analysis of the full productive capacity of the animal.

In the same way, the early estimate of the lactation curve can be used to predict the total production that a sheep is going to generate during the milking period, with no need to wait until this period ends. At the same time it permits the identification of the most productive sheep in the flock that should be destined for reproduction, thus contributing to the genetic progress of the flock and, consequently, to an increase in profits for the enterprise under analysis.

Results obtained with those network models in which genetic and pruning algorithms are applied were found to be better than other models which exclusively applied the classical learning algorithm (the EDBD algorithm). This occurred especially in the specification of the *c* parameter, whose distribution of values was much more homogeneous. Probably the bigger size of the neural network in EDBD application (without pruning) generated over-training and reduced the generalization capacity of the network in this case. In general, we recommended the use of GA with pruning algorithms always that the network structure is unknown. In this case is better to start out with a network of a considerable size and to gradually eliminate the nodes and unnecessary weights until reaching a network size that provided satisfactory results.

The results obtained with ANN models were better than the ones obtained with regression models (LR and QR).

Despite the loss of information implied by working with production data averaged by fortnights (due to the lack of regularity in the time spans in which the controls were registered), and the difficulties entailed by the lack of homogeneousness in the *A*, *b* and *c* parameter values defining the lactation curve of each sheep, the results obtained in the definition of the lactation curve parameters seem to be quite acceptable. This demonstrates the potential use of this methodology, taking into account that better quality data could readily turn out considerably superior results.

This research has attempted to demonstrate how these new artificial intelligence techniques can be employed to resolve situations involving forecasting similar to those that arise in business management in general, and more specifically, in livestock enterprises. The use of these techniques presents an interesting alternative when initial hypotheses necessary for the application of standard multivariate statistical techniques are not fulfilled.

Acknowledgements

The authors gratefully acknowledge the financial support provided by the Spanish Department of Research of the Ministry of Science and Technology under the TIC2002-04036-C05-02 (Department of Computer Science, University of Cordoba) Projects. FEDER also provided additional funding.

References

[1] White H. Economic prediction using neural networks: the case of IBM daily stock returns. Neuralnetworks in Finance and Investing 1993; 315–29.
[2] Bosch J, Garrido LC. Predicción de índices de futuros financieros mediante redes neuronales. Swaps & Productos Derivados 1997;27:19–21.
[3] Curry B, Morgan P. Neural networks and non-linear statistical methods: an application to the modeling of price-quality relationships. Computers & Operations Research 2002;29:951–69.
[4] An-Sing Ch, Leesy M. Forecasting exchange rates using general regression neural networks. Computers & Operations Research 2000;27:1093–110.
[5] Werbos PJ, Beyond regression: new tools for prediction and analysis in the behavioural sciences. Ph.D. thesis, Harvard University, Cambridge, 1974.
[6] Rumelhart DE, Mcdelland JC. Parallel distributed processing: explorations in the microstructure of cognition. Cambridge, MA: MIT Press; 1986, p.1.
[7] Widrow B. Layered neural nets for pattern recognition. IEEE 1962;36:1109–18.
[8] Williams RJ, Minai AA. Back-propagation heuristics: a study of the Extended Delta-Bar-Delta algorithms. IEEE International Joint Conference on Neural Networks, vol. 1. 1990. p. 595–600.
[9] Jacobs RA. Increased rates of convergence trough learning rate adaptation. Neural Networks 1978;1:295–307.
[10] Hecht-Nielsen R. Kolmogorov mapping neural network existence theorem. First IEEE International Conference on Neural Network, vol. 3. 1987. p. 11–4.
[11] Carrol B, Dickinson BD. Construction of neural nets using the randon transform. Proceedings of the International Joint Conference on Neural Networks, vol. I. 1989. p. 607–11.
[12] Cybenko G. Approximation by superpositions of a sigmoidal function. Control Signals Systems 1989;2:305–14.
[13] Stinchcombe M, White H. Approximating and learning unknown mappings using multilayer feedforward networks with bounded weight. Proceedings of the International Joint Conference on Neural Networks, vol. III. 1990. p. 7–15.
[14] Hornik K, Stinchcombe M, White H. Multi-layer feedforward networks are universal approximators. Neural networks 1989;2:359–66.
[15] Barron AR. Universal approximation bounds for superpositions of a sigmoid function. IEEE Transactions on Information Theory 1993;39:930–45.
[16] Bebis M, Georgipoulos M. Coupling weight elimination with genetic algorithms to reduce network size and preserve generalization. Neurocomputing 1997;17:167–94.
[17] Williams PM. Bayesian regularization and pruning using a Laplace Prior. Neural Computation 1994;7:117–43.
[18] Honaver V, Balakrishnan K. Evolutionary design of neural architectures. A preliminary taxonomy and guide to literature. 1998.
[19] Whitley LD, Schaffer JD. International workshop on combinations of genetic algorithms and neural networks. Silverspring, MD: IEEE Computer Society Press; 1992.

[20] Angeline PJ, Saunders GM, Pollack JB. An evolutionary algorithm that constructs recurrent neural networks. IEE Transactions on Neural Networks 1994;5(1).
[21] Yao X. Envolving artificial neural networks. Proceedings of the IEEE 1999;9(87):1423–47.
[22] García N, Hervás C. Symbiotic: cooperative coevolution of neural networks. IEE Transactions on Neural Networks, in press.
[23] Buxadé C. Ovino de leche: aspectos clave. Madrid: Mundi Prensa; 1998.
[24] Montoro V, Pérez-Guzmán MD. La selección de la raza ovina Manchega. Junta de Comunidades de Castilla la Mancha 1996, No.9.
[25] Molina MP. Composición y factores de variación de la leche de ovejas de raza Manchega. Tesis doctoral Universidad de Valencia, España, 1987.
[26] Whiting R, Buchanan R. Microbial Modelling. Food Technology 1994;48:113–20.
[27] Treacher TT, Gribb MJ. The milk yield of Finnish Landrace x Dorset Horn ewes milked by machine. II Symposium International sur la traite mécanique des petists rumiants, 1978. p. 113–22.
[28] Carriedo JA, San Primitivo F. Estudio genético de los factores que influyen en la producción láctea de ganado ovino II. Factores ambientales. ITEA 1983;47:29–34.
[29] Gallego L, Bernabeu R. Producción de leche: factores de variación en ganado ovino de raza Manchega. Madrid: Mundi Prensa; 1994. p. 162–73.
[30] Morant SV. A new approach to the formulation of lactarion curves. Animal Production 1989;49:151–62.
[31] Munakatam T. Fundamentals of the new artificial intelligence. New York: Springer; 1998.
[32] Mackay DJC. Bayesian interpolation. Neural Computation 1992;4:415–47.
[33] Mackay DJC. A practical Bayesian framework for backpropagation networks. Neural Computation 1992;4:448–72.
[34] Goldberg DE. Genetic algorithms in search optimization and machine learning. Reading, MA: Addison-Wesley; 1989.
[35] Judge G, Hill C, Griffiths W. Introduction to the theory and practice of econometrics. New York: Wiley; 1982. p. 619.
[36] Serrano M, Montoro V. Cálculo de los factores de extensión de la lactación a 120 días en ganado ovino Manchego. Investigación agraria: producción y Sanidad Animales 1996; 11(1): 69–83.