ELSEVIER

Food Microbiology 24 (2007) 452–464

FOOD MICROBIOLOGY

www.elsevier.com/locate/fm

Product unit neural network models for predicting the growth limits of *Listeria monocytogenes*

A. Valero[a,*], C. Hervás[b], R.M. García-Gimeno[a], G. Zurera[a]

[a]*Department of Food Science and Technology, University of Cordoba, Campus Rabanales, Edif. Darwin, 14014 Córdoba, Spain*
[b]*Department of Computer Science and Numerical Analysis, University of Cordoba, Campus Rabanales Edif. A. Einstein, 14014 Córdoba, Spain*

Received 30 June 2006; received in revised form 23 October 2006; accepted 27 October 2006
Available online 8 December 2006

Abstract

A new approach to predict the growth/no growth interface of *Listeria monocytogenes* as a function of storage temperature, pH, citric acid (CA) and ascorbic acid (AA) is presented. A linear logistic regression procedure was performed and a non-linear model was obtained by adding new variables by means of a Neural Network model based on Product Units (PUNN). The classification efficiency of the training data set and the generalization data of the new Logistic Regression PUNN model (LRPU) were compared with Linear Logistic Regression (LLR) and Polynomial Logistic Regression (PLR) models. 92% of the total cases from the LRPU model were correctly classified, an improvement on the percentage obtained using the PLR model (90%) and significantly higher than the results obtained with the LLR model, 80%. On the other hand predictions of LRPU were closer to data observed which permits to design proper formulations in minimally processed foods. This novel methodology can be applied to predictive microbiology for describing growth/no growth interface of food-borne microorganisms such as *L. monocytogenes*. The optimal balance is trying to find models with an acceptable interpretation capacity and with good ability to fit the data on the boundaries of variable range. The results obtained conclude that these kinds of models might well be very a valuable tool for mathematical modeling.

Keywords: Logistic regression; Classification; Product unit neural networks; Evolutionary computation; Boundary models; *Listeria monocytogenes*

1. Introduction

In the two last decades, models were developed in the field of predictive microbiology to evaluate the behavior of microorganisms under a given set of conditions.

Recently, however, due to the demand for more healthy and convenient food products, scientists recognize that there is an increasing need to model microbial growth limits (McMeekin et al., 2000). Growth/no growth models or boundary models quantify the probability of microbial growth and define combinations of factors that prevent growth. This is because microbial growth is confined to a limited range of factors, and sometimes growth even drops sharply when the level of each factor is increased. The importance of growth boundary models for empowering the hurdle concept has been discussed by various authors (Schaffner and Labuza, 1997; McMeekin et al., 2000). Therefore, these kinds of models can be useful for the development of processes that allow safer food products to be produced and provide information about more realistic estimations of food safety risks. In addition, they might be important in decisions regarding food safety regulations (Schaffner and Labuza, 1997).

Several of these models have been defined over the last few years: Ratkowsky and Ross (1995) developed a relationship between growth probability and a polynomial function derived from a secondary square-root model, ln-transformed, as the basis of the growth boundary model using linear logistic regression. This approach was followed by other authors (Lanciotti et al., 2001) and predicts a binary response variable, or equivalently the probability of an event's occurrence in terms of a specific set of explicative variables related to it. Later on, a new nonlinear logistic regression technique has been performed (Salter et al., 2000; Tienungoon et al., 2000). Logistic models were

*Corresponding author. Tel.: +34 957 21 20 57; fax: +34 957 210 000.
E-mail address: bt2vadia@uco.es (A. Valero).

0740-0020/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.fm.2006.10.002

further developed to determine growth/no growth interfaces in food-based systems (McKellar et al., 2002).

The logistic regression constitutes a simple and useful procedure, although it poses problems when is applied to a real-problem of classification, where frequently we cannot make the stringent assumption of additive and purely linear effects of the variables. A traditional technique to settle this problem is introducing additional variables, basis functions, which are transformations of the input variables and then to use linear models in this new space of derived input features. The advantage of this method is that once the basis functions have been determined, the models are linear in these new variables and the fitting follows a standard procedure. This paper presents a new approach to determine the growth probability of *Listeria monocytogenes* using logistic regression where the linear functions are made up of the input variables and transformations by training Product Unit Neural Network models (PUNN). The interpretability should not be obtained at the cost of drastically decreasing, either the good degree of prediction capacity or robustness. This type of model can be proposed in predictive microbiology since it is logical to suppose a priori that a strong interaction exists between the factors that affect the prediction of microbial growth parameters. Thus the use of PUNN has two major advantages: these product units are more effective in detecting interactions between the factors and they are easier to interpret than other Artificial Neural Network (ANN) models. The product units have the ability to implement higher-order functions and therefore they can also implement polynomial functions as a particular case (Gurney, 1992). PUNN models are formed by linear combinations of functions of a potential type that are not as smooth as sigmoidal type functions. This characteristic enables PUNN models to tackle complex decision-making more easily.

Very few reports in the literature in the field of predictive microbiology have evaluated the performance of different logistic regression methods to define the microbial growth probability. Hajmeer and Basheer (2003a) performed a research work based on the data from Salter et al. (2000) regarding the growth/no growth of *Escherichia coli R31* as a function of temperature and water activity. They compared techniques based on ANN and found that the Probability Neural Networks models classified the data better than the Feed-forward Error Back-propagation Artificial Neural Network (FEBANN). In that study, in relation to logistic models, the quadratic model was more accurate than the linear one.

In this paper, different logistic regression techniques were performed and compared to describe the growth probability of *L. monocytogenes* as a function of storage temperature, pH, citric acid (CA) and ascorbic acid (AA): Lineal Logistic Regression model (LLR), Polynomial Logistic Regression model (PLR)) and a new approach of logistic regression models using non-linear terms obtained by Product Unit Neural Network models (LRPU). The accuracy of various classifiers was assessed and compared, in addition to the advantages and limitations of each technique.

2. Material and methods

2.1. Strain and culture conditions

From a slant culture, a pellet containing *L. monocytogenes* (strain NCTC 11994) was cultured during 24 h at 30 °C in 10 ml of pure Brain Heart Infusion (BHI) (Oxoid CM225 ltd., Basingstoke, Hampshire, UK) Afterwards, 1 ml of the culture was transferred to 9 ml and sub-cultured twice (24 h, 30 °C each). A third subculture was obtained, incubating 1 ml of the inoculum in a flask of 100 ml of BHI until the early stationary phase was reached.

2.2. Experimental design

A fractional factorial design was followed in order to find out the growth limits of *L. monocytogenes*. Data were collected at CA and AA concentrations of 0, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35 and 0.4 % (w/v), at 4, 7, 10, 15 and 30 °C and pH levels of 4.5, 5, 5.5 and 6. 539 different conditions were tested with eight replicates per condition. This data set was divided so that 305 conditions covering the extreme domain of the model were chosen for training, and 234 conditions were selected inside the range of the model for testing the generalization capacity (Table 1). To determine which data belong to model training and model validation, we selected alternatively the conditions of organic acids used at the same level of temperature and pH, as shown in Table 2. The purpose of this selection was to define data sets for model training that really represent the boundary zones in order to obtain a better fit.

The quantity of acids to be added was calculated in percentages (w/v) in order to imitate the addition of organic acids that takes place in food industries. Previous examination of the regression analysis revealed that the use of undissociated acid concentration (u.a.c) as an independent factor instead of the total acid concentration resulted in more accurate models. This is a reasonable observation, as it has been clearly demonstrated that u.a.c of organic acids is the effective molecule that causes inhibition of growth (ICMSF, 1980). Therefore, CA and AA factors were included in the analysis in terms of u.a.c which was calculated using the Henderson–Hasselbach equation:

For citric acid,

$$\text{u.a.c} = \frac{\text{Ca}(\text{H}^+)^3}{(\text{H}^+)^3 + K_1(\text{H}^+)^2 + K_1K_2(\text{H}^+)^1 + K_1K_2K_3}. \quad (1)$$

For ascorbic acid,

$$\text{u.a.c} = \frac{\text{Ca}(\text{H}^+)^2}{(\text{H}^+)^2 + K_1(\text{H}^+)^1 + K_1K_2}, \quad (2)$$

Table 1
Conditions selected for model training and model validation for the environmental factors considered

T (°C)	pH	CA (% w/v)	AA (% w/v)
30	6	0	0.4
		0.4	0
		0	0
		0.4	0.4
	5.5	0	0
		0	0.4
		0.35	0–0.4
		0.4	0–0.4
	5	0	0
		0	0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0–0.4
	4.5	0	0
		0	0.4
		0.15	0–0.4
		0.2	0–0.4
		0.25	0–0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0–0.4
15	6	0	0.4
		0.4	0
		0	0
		0.4	0.4
	5.5	0	0
		0	0.4
		0.25	0–0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0–0.4
	5	0	0
		0.4	0
		0.25	0–0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0
		0.4	0.4
	4.5	0	0–0.4
		0.05	0–0.4
		0.4	0.4
		0.4	0
10	6	0	0.4
		0.4	0
		0	0
		0.4	0.4
	5.5	0	0
		0	0.4
		0.25	0–0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0–0.4
	5	0	0
		0	0.4
		0.1	0–0.4
		0.15	0–0.4
		0.2	0–0.4
		0.25	0–0.4
		0.3	0–0.4
		0.4	0
		0.4	0.4
	4.5	0	0.4
		0.4	0
		0	0
		0.4	0.4
7	6	0	0
		0	0.4
		0.3	0–0.4
		0.4	0–0.4
	5.5	0	0
		0	0.4
		0.2	0–0.4
		0.25	0–0.4
		0.3	0–0.4
		0.35	0–0.4
		0.4	0–0.4
	5	0	0
		0	0.4
		0.05	0–0.4
		0.1	0–0.4
		0.15	0–0.4
		0.2	0–0.4
		0.25	0–0.4
		0.3	0–0.4
	4.5	0	0.4
		0.4	0
		0	0
		0.4	0.4
4	6	0	0
		0	0.4
		0.3	0–0.4
		0.4	0–0.4
	5.5	0	0
		0	0.4
		0.15	0–0.4
		0.2	0–0.4
		0.25	0–0.4
		0.3	0–0.4
		0.4	0
		0.4	0.4
	5	0	0–0.4
		0.05	0–0.4
		0.1	0–0.4
		0.15	0–0.4
		0.2	0–0.3
		0.25	0–0.3
		0.4	0
		0.4	0.4
	4.5	0	0.4
		0.4	0
		0	0

where Ca is the total concentration of the acid, expressed in mM, H^+ the proton concentration (mM) at a certain pH, and K_x are the dissociation constants of the organic acids, which were recalculated to mM to incorporate in Eqs. (1) and (2) ($K_1 = 0.00072$; $K_2 = 1.7 \times 10^{-5}$; $K_3 = 4.1 \times 10^{-7}$ for citric acid, and $K_1 = 0.0001$; $K_2 = 1.6 \times 10^{-12}$ for ascorbic acid). Organic acids employed are expressed in percentage (w/v), total concentration (t, mmol/l), undissociated acid

Table 2
Experimental design followed at a same level of temperature and pH

CA (%)	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
AA (%)									
0	○	◆	○	◆	○	◆	○	◆	○
0.05	◆	○	◆	○	◆	○	◆	○	◆
0.1	○	◆	○	◆	○	◆	○	◆	○
0.15	◆	○	◆	○	◆	○	◆	○	◆
0.2	○	◆	○	◆	○	◆	○	◆	○
0.25	◆	○	◆	○	◆	○	◆	○	◆
0.3	○	◆	○	◆	○	◆	○	◆	○
0.35	◆	○	◆	○	◆	○	◆	○	◆
0.4	○	◆	○	◆	○	◆	○	◆	○

CA = Citric acid; AA = ascorbic acid.
○, training data; ◆, validation data.

concentration (u.a.c, mM) and undissociated acid fraction (u.a.f) in Table 3.

2.3. Growth medium preparation

Different concentrations of citric acid L (−) L-hydrate (Panreac 131018, Barcelona Spain) and L (+) ascorbic acid (Panreac 131013, Barcelona Spain) were added to 100 ml of BHI broth to achieve the desired combinations of both acids. The pH of the media was adjusted with solutions of 1 M of HCl and NaOH (Panreac, Barcelona, Spain) brought to its final volume, and the final pH was checked (pH/mv-meter digit 501, Crison, Barcelona, Spain). All the modified media were sterilised through 0.22 μm sterile filters (Millipore, Madrid, Spain) and were kept in refrigeration until inoculation.

2.4. Inoculation procedure

The procedure described by Carrasco et al. (2006) was followed. A calibration equation was performed in order to determine the number of cells inoculated into the media by gathering three data sets from three previous calibrations in which viable counts were plotted against OD data obtained in Bioscreen C (Labsystems, Finland). At the same time the inoculum size was checked by plate count on Tryptone Soya Agar (TSA) (Oxoid ltd., Basingstoke, Hampshire, UK) and incubated at 30 °C for 24 h.

From the inoculum grown in BHI, the necessary dilutions in buffered 0.1% peptone water (Oxoid Ltd., Basingstoke, Hampshire, UK) were carried out, to obtain a concentration of 5×10^5 CFU/ml. Then, eight replicate microtiter wells were filled up with 160 μl of the modified media, and inoculated with 40 μl, reaching a final concentration of 10^5 CFU/ml per well. Two wells per condition studied served as controls (un-inoculated medium). Afterwards, microtiter plates were sealed with paraffin and refrigerated at the appropriate temperatures.

2.5. Growth/no growth evaluation

Growth was monitored by absorbance measurements in Bioscreen C (Labsystems, Finland) during 21 days at all conditions. Preliminary studies performed on Bioscreen C, showed that a significant change of turbidity was detected (and consequently growth) by an increase of the absorbance of 0.2 units. By means of the calibration equation, optical density data were transformed into plate counts. The absorbance of the un-inoculated medium, around 0.15 units, was subtracted to the absorbance values of the inoculated media before applying calibration curves. In this study, growth was considered by direct turbidimetric measurements when the cell density reached 7.5 log CFU/ml, corresponding to an absorbance value of 0.350 approximately. However, to solve the problem of the detection limit of Bioscreen C, for the closest samples near the boundary zone (which gave absorbance values less than 0.350), the bacterial population in the well was determined by direct plating on TSA and comparing with the initial inoculum size. Growth was considered if a difference of more than 1 log CFU/ml with the initial inoculum was detected. Growth/no growth evaluation by turbidimetric measurements was followed in other studies (Lanciotti et al., 2001; Koutsoumanis et al., 2004) concluding good results for predicting microbial interface.

For any combination of factors, probability of growth (P) was recorded as "1" if it occurred and "0" if did not. The predicted growth/no growth interfaces given by the PLR and the LRPU models for $P = 0.9$, 0.5 and 0.1 were calculated using Microsoft Excel Solver.

To start processing data, each of the input and output variables were scaled in the ranks [0.1, 0.9] and [1, 2], respectively. The lower bound is chosen to avoid inputs values near to 0 that can produce very large values of the function for negative exponents. The upper bound is chosen near 1 to avoid dramatic changes in the outputs of the network when there are weights with large values (especially in the exponents). The new scaled variables were named T^*, pH*, CA* and AA*, to the input variables and $G^* = G + 1$ for the output variable, that is $G^* \in \{1, 2\}$. For example, T is calculated as follows:

$$T^* = \frac{T - T_{\min}}{T_{\max} - T_{\min}} \times 0.8 + 0.1, \quad (3)$$

where T is the original temperature, $T_{\min}$ and $T_{\max}$, are the minimum and maximum values, and T^* is the scaled temperature. The model can be applied by de-scaling the variables in the same form.

2.6. Logistic regression models

We consider the situation where we observe a binary outcome variable $\mathbf{y}$ and a vector $\mathbf{x} = (1, x_1, x_2, \ldots, x_k)$ of covariates for each N individuals (we assume that the vector of inputs includes the constant term 1 to

Table 3
Organic acids employed are expressed in percentage (w/v), total concentration (t, mmol/l), undissociated acid concentration (u.a.c., mmol/l) and undissociated acid fraction (u.a.f.)

CA (%)	AA (%)	pH	CA (*t*)	AA (*t*)	CA (u.a.c.)	AA (u.a.c.)	CA (u.a.f.)	AA (u.a.f.)
0	0	4.5	0	0	0	0	0	0
0.05	0.05		2.604	2.840	0.072	0.682	0.027	0.240
0.1	0.1		5.208	5.681	0.144	1.365	0.027	0.240
0.15	0.15		7.812	8.522	0.216	2.047	0.027	0.240
0.2	0.2		10.416	11.363	0.288	2.730	0.027	0.240
0.25	0.25		13.020	14.204	0.360	3.412	0.027	0.240
0.3	0.3		15.625	17.045	0.432	4.095	0.027	0.240
0.35	0.35		18.229	19.886	0.504	4.777	0.027	0.240
0.4	0.4		20.833	22.727	0.576	5.460	0.027	0.240
0	0	5	0	0	0	0	0	0
0.05	0.05		2.604	2.840	0.012	0.258	0.004	0.090
0.1	0.1		5.208	5.681	0.025	0.516	0.004	0.090
0.15	0.15		7.812	8.522	0.038	0.774	0.004	0.090
0.2	0.2		10.416	11.363	0.051	1.033	0.004	0.090
0.25	0.25		13.020	14.204	0.064	1.291	0.004	0.090
0.3	0.3		15.625	17.045	0.077	1.549	0.004	0.090
0.35	0.35		18.229	19.886	0.090	1.807	0.004	0.090
0.4	0.4		20.833	22.727	0.103	2.066	0.004	0.090
0	0	5.5	0	0	0	0	0	0
0.05	0.05		2.604	2.840	0.001	0.087	0.0006	0.030
0.1	0.1		5.208	5.681	0.003	0.174	0.0006	0.030
0.15	0.15		7.812	8.522	0.004	0.261	0.0006	0.030
0.2	0.2		10.416	11.363	0.006	0.348	0.0006	0.030
0.25	0.25		13.020	14.204	0.008	0.435	0.0006	0.030
0.3	0.3		15.625	17.045	0.009	0.522	0.0006	0.030
0.35	0.35		18.229	19.886	0.011	0.609	0.0006	0.030
0.4	0.4		20.833	22.727	0.012	0.696	0.0006	0.030
0	0	6	0	0	0	0	0	0
0.05	0.05		2.604	2.840	0.0001	0.028	5.561E-05	0.009
0.1	0.1		5.208	5.681	0.0002	0.056	5.561E-05	0.009
0.15	0.15		7.812	8.522	0.0004	0.084	5.561E-05	0.009
0.2	0.2		10.416	11.363	0.0005	0.112	5.561E-05	0.009
0.25	0.25		13.020	14.204	0.0007	0.140	5.561E-05	0.009
0.3	0.3		15.625	17.045	0.0008	0.168	5.561E-05	0.009
0.35	0.35		18.229	19.886	0.0010	0.196	5.561E-05	0.009
0.4	0.4		20.833	22.727	0.0011	0.225	5.561E-05	0.009

accommodate the intercept). We coded the two-class via a 0/1 response y_i, where $y_i = 1$ for the first class (growth) and $y_i = 0$ for the second class (no growth). Let p be the conditional probability associated with the first class. Logistic regression is a widely used statistical modeling technique in which the probability p of the dichotomous outcome event is related to a set of explanatory variables x in the form:

$$\text{logit}(p) = \ln\left(\frac{p}{1-p}\right) = f(\mathbf{x},\boldsymbol{\beta}) = \boldsymbol{\beta}^{\mathrm{T}}\mathbf{x} = \boldsymbol{\beta}_0 + \beta_1\mathbf{x}_1 + \cdots + \boldsymbol{\beta}_k\mathbf{x}_k \quad (4)$$

where $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \ldots, \beta_k)$ is the vector of the coefficients of the model and $\boldsymbol{\beta}^{\mathrm{T}}$ the transpose vector. We refer to $p/1-p$ as odds-ratio as the log-odds or logit transformation. A simple calculation in Eq. (4) shows that the probability of occurrence of an event as a function of the initial variables is non-linear and is given by

$$p(x;\beta) = \frac{e^{\boldsymbol{\beta}^{\mathrm{T}}\mathbf{x}}}{1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\mathbf{x}}}. \quad (5)$$

The decision boundary is the set of points for which the log-odd are zero, that is, in this lineal model, a hyperplane defined by $\{x|\boldsymbol{\beta}^{\mathrm{T}}\mathbf{x} = 0\}$. Observe that logistic regression not only constructs a decision rule but also finds a function that for any input vector defines the probability p that the vector $\mathbf{x}$ belongs to the first class.

However, as described above, it is useful to amplify/replace the vector of inputs X with additional variables, which are transformations of X, and then use lineal models in this new space of derived input features.

A first approximation to these kinds of models is to define a Polynomial Logistic Regression model, PLR, where the cross-product terms and the quadratic effects are

included in the analysis; so Eq. (4) is modified as

$$\text{logit}\,(p) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \beta_{12} X_1 X_2 + \cdots + \beta_{k-1,k} X_{k-1} X_k + \beta_{1,1} X_1^2 + \cdots + \beta_{k,k} X_K^2. \quad (6)$$

The use of functions $f(\mathbf{x},\boldsymbol{\beta})$ of a quadratic response surface is not justified in all cases, since the polynomial order cannot be determined a priori and different tests would have to be performed with different polynomial orders. If the main reason to build polynomial functions is to attempt to predict growth probability as a function of environmental factors and their possible interactions, in a search for more flexible models, a neural network model based on functions of potential base might be an alternative option to consider.

In this paper, we applied a new logistic regression model based on the combination of the standard linear model and product-unit models, introducing a non-linear term in the model constructed with basis functions given by products of the inputs raised to real powers, which express the possible strong interactions among the factors. The methodology was based on a previous study of Hervás-Martínez and Martínez-Estudillo (2006). The general expression of the $f(\mathbf{x}, \boldsymbol{\theta})$ is given by

$$f(\mathbf{x}, \boldsymbol{\theta}) = \alpha_0 + \sum_{j=1}^{k} \alpha_i x_i + \sum_{j=1}^{m} \beta j \prod_{i=1}^{k} x_i^{w_{ji}}. \quad (7)$$

With the corresponding matricial expression

$$f(\mathbf{x}, \boldsymbol{\theta}) = \alpha^{\mathrm{T}} x + \boldsymbol{\beta}^{\mathrm{T}} \mathbf{B}(\mathbf{x}, \mathbf{W}), \quad (8)$$

where the basis functions are $\mathbf{B}(\mathbf{x}, \mathbf{W}) = (B_1(\mathbf{x},\mathbf{w}_1), B_2(\mathbf{x},\mathbf{w}_2),\ldots, B_m(\mathbf{x},\mathbf{w}_m))$ with $B_j(\mathbf{x},\mathbf{w}_j) = \prod_{i=1}^{k} x_i^{w_{ji}}$ and the parameters $\boldsymbol{\theta} = (\alpha, \boldsymbol{\beta}, \mathbf{W})$, $\boldsymbol{\alpha}$ $(\alpha_0, \alpha_1,\ldots, \alpha_k)$, $\boldsymbol{\beta} = (\beta_0,\beta_1,\ldots, \beta_m)$ and $\mathbf{W} = (\mathbf{w}_1,\ldots,\mathbf{w}_j)$ where $\mathbf{w}_j = (w_{j1},\ldots,w_{jp})$ with $w_{\underline{ji}} \in R$.

In this way the new set of conditional distributions are:

$$p(\mathbf{x}, \boldsymbol{\beta}) = \frac{e^{f(x,\theta)}}{1 + e^{f(x,\theta)}} \quad (9)$$

and the logit transformation

$$\text{logit}(p) = \ln\left(\frac{p}{1-p}\right) = f(x,\theta). \quad (10)$$

In this case the decision boundaries are non-linear and defined by the hypersurface $f(\mathbf{x},\boldsymbol{\theta}) = 0$ in the $\mathbb{R}^k$ space. The non-linear part $f(\mathbf{x},\boldsymbol{\theta})$ corresponds to a special class of feed-forward neural networks, namely PUNN, introduced by Durbin and Rumelhart (1989) and recently developed by Martínez-Estudillo et al. (2006) and based on multiplicative nodes instead of the additive ones.

In our framework, the product units have the following structure: an input layer with a node for every input variable, a hidden layer with several nodes and an output layer with just one node (to classify the different results into two classes: growth and no growth). There are no connections between the nodes in a layer and none between the input and output layers either. The network has k inputs that represent the independent variables of the model, m nodes in the hidden layer and one node in the output layer. The activation of the jth node in the hidden layer is given by $B_j(x, w_j) = \prod_{i=1}^{k} x_i^{w_{ij}}$ where w_{ij} is the connection between input node i and hidden node j. The activation of the output node is given by $\beta_0 + \sum_{j=1}^{m} \beta_j B(x, w_j)$ where β_j is the connection between the hidden node j and the output node. The transfer function of all nodes is the identity function.

2.7. Estimation of the coefficients

The methodology proposed was based on the combination of an evolutionary algorithm, EA, (global explorer) and a local optimization procedure (local exploiters) carried out by standard maximum likelihood optimization method. In a first step, an EA was applied to design the structure and training the weights of a product-unit neural network. The evolutionary process determines the number m of potential basis functions of the model and the corresponding vector $\mathbf{w}_j$ of exponents. Once the basis functions $B(x, W) = (B_1(x,w1), B_2(x,w_2),\ldots, B_m(x,w_m))$ have been determined by the EA, we considered a transformation of the input space adding the non-linear transformations of the input variables given by the basis functions obtained by the evolutionary algorithm. The remaining coefficients were calculated by the maximum likelihood optimization method. Finally, we used a backward stepwise procedure, pruning variables sequentially to the model obtained previously, until further prunes did not improve the fit.

In the following paragraphs we describe the methodology recently developed by Hervás-Martínez et al. (2006).

2.8. Hybrid estimation methodology

We applied initially an evolutionary algorithm, EA, to find the basis functions $\mathbf{B}(\mathbf{x}, \mathbf{W}) = \{B_1(\mathbf{x}, \mathbf{w}_1), B_2(\mathbf{x}, \mathbf{w}_2), \ldots, B_m(\mathbf{x}, \mathbf{w}_m)\}$ corresponding to the non-linear part of $f(\mathbf{x},\boldsymbol{\theta})$. In this step, we used an EA to train the PUNN. The training procedure consists of estimating the weight connections, w_{ji}, of the input layer with a hidden layer of a PUNN, and the number, m, of the basis functions to the net. The search begins with an initial population where for each iteration the population is updated using a population-update algorithm. The population is subject to the operations of replication and mutation.

The general structure of the EA is described in detail in Martínez-Estudillo et al. (2006), and it can be supported in the following steps:

(a) Generate initial population (N_R) with randomly generated networks.
(b) Evaluate the fitness score for each individual of the population based on the objective function.

(c) Copy the best individual to the next generation.
(d) The best 10% of the population substitutes the worst 10% of individuals.
(e) Apply parametric mutation operators to the best 10% of the population.
(f) Apply structural parametric mutation to the rest of the population (90%).

Parametric mutation consists of a simulated annealing algorithm (Kirkpatric et al., 1983).

For determining the goodness of a model g we considered the mean squared error (MSE) of that individual g of the population:

$$\mathrm{MSE}(g) = \frac{1}{n_T}\sum_{l=1}^{n_T}(y_l - g(\mathbf{x}_l))^2, \tag{11}$$

where y_1 are the predicted values. We defined the fitness function $A(g)$ by means of a strictly decreasing transformation of the mean squared error in Eq. (11):

$$A(g) = \frac{1}{1+\mathrm{MSE}(g)}. \tag{12}$$

The severity of a mutation to an individual g is dictated by the temperature $T(g)$, given by $T(g) = 1–A(g)$, $0 \leqslant T(g) < 1$.

Structural mutation implies a modification of the structure of the function and allows the explorations of the different regions of the search space and helps to keep the diversity of the population. There are four different structural mutations: node addition (AN), node deletion (DN), connection addition (AC) and connection deletion (DC). All the above mutations are made sequentially in the given order, with probability $T(g)$ in the same generation on the same network. If the probability does not select any mutation, one of the mutations is chosen at random and applied to the network. The stop criterion is reached when there is no improvement either in the average of the performance of a percentage of the best individuals in the population, or in the fitness of the best individual, or when the algorithm achieves a determined number of generations. In the next step the input space was transformed by adding the non-linear transformations of the input variables given by the basis functions obtained by the EA.

$H : \mathbb{R}^k \to \mathbb{R}^{k+m}$

$(x_1, x_2, \ldots, x_k) \to (x_1, x_2, \ldots, x_k, z_1 \ldots, z_m)$,

where $z_1 = B_1(x, \hat{w}_1) \ldots\ldots z_m = B_m(x, \hat{w}_m)$.

Following the proposed methodology, a (standard) maximum likelihood estimation method was applied in the new space of derived input features. The log-likelihood function for n_T observations was optimized as follows:

$$l(\boldsymbol{\theta}^1) = \sum_{i=1}^{n_T}\left\{ y_i f(\mathbf{x},\boldsymbol{\theta}^1) - \log(1 + e^{f(\mathbf{x},\boldsymbol{\theta}^1)})\right\}, \tag{13}$$

where $\boldsymbol{\theta}^1 = (\alpha, \beta, \hat{w})$. The estimated coefficients $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\beta}}, \hat{\mathbf{w}})$ are obtained by means of the Newton–Raphson optimization algorithm. The estimation process determines the model:

$$f(\mathbf{x}, \hat{\boldsymbol{\theta}}) = \hat{\alpha}_0 + \sum_{i=1}^{k}\hat{\alpha}_i x_i + \sum_{j=1}^{m}\hat{\beta}_j \prod_{i=1}^{k} x_i^{\hat{w}_{ji}}. \tag{14}$$

Finally, in order to select the final model, a backward stepwise procedure was used, starting with the full model and successively pruning variables sequentially to the model until further prunes do not improve the fit. At each step, the least significant parameters to predict the response variable were deleted, that is, the one which showed the greatest critical value (p-value) in the hypothesis test, where the associated coefficient equal to zero is the hypothesis to be contrasted. The procedure finished when all tests provided p-values smaller than the fixed significance level and the model selected in the previous step fitted well.

2.9. Statistical analysis

Both Lineal and Polynomial Logistic Regression models were fitted to the growth/no growth data obtained by means of a regression modeling procedure of SPSS for Windows 11.0 (SPSS, 2003 Inc., Chicago, IL, USA). The regression was performed with a forward conditional method, which selects the most significant factors. To measure the classifier's performance, the output was compared to the observed outcome (event or non-event), and assigned one of the four possible situations: (i) true negative (TN) when negative cases are correctly identified by the classifier; (ii) false positive (FP) when the classifier incorrectly identifies a non-event case as an event; (iii) false negative (FN) when the classifier incorrectly identifies an event case as a non-event; and (iv) true positive (TP) when the positives cases are correctly identified. For a given number of cases (N) these index are inserted into a contingency matrix (C-matrix) as

$$C - \mathrm{matrix} = \begin{pmatrix} N_{\mathrm{TN}} & N_{\mathrm{FP}} \\ N_{\mathrm{FN}} & N_{\mathrm{TP}} \end{pmatrix},$$

where $N = N_{\mathrm{TN}}+N_{\mathrm{FP}}+N_{\mathrm{FN}}+N_{\mathrm{TP}}$. The sum of $N_{\mathrm{TN}}+N_{\mathrm{TP}}$ is the total number of cases that were classified correctly, whereas $N_{\mathrm{FN}}+N_{\mathrm{FP}}$ is the total number of misclassified cases.

The performance measures used were those described by Hajmeer and Basheer (2003a): (i) fraction correct, FC, representing the proportion of cases classified correctly; (ii) false alarm (positive) rate, FAR, which represents the fraction of negative cases classified as positives; and (iii) probability of detection, POD, representing the fraction of true cases detected as positives by the classifier. These concepts are expressed as follows:

$$\mathrm{FC} = \frac{N_{\mathrm{TN}}+N_{\mathrm{TP}}}{N}, \quad \mathrm{FAR} = \frac{N_{\mathrm{FP}}}{N_{\mathrm{TN}}+N_{\mathrm{FP}}},$$
$$\mathrm{POD} = \frac{N_{\mathrm{TP}}}{N_{\mathrm{FN}}+N_{\mathrm{TP}}}. \tag{15}$$

For a perfect classifier, FC = POD = 1 and FAR = 0.

3. Results and discussion

3.1. Evaluation of the goodness of fit

Among the different conditions tested, there were 240 no growth cases and 299 growth cases. A forward conditional method was selected in SPSS in order to obtain the significant variables of the logistic regression model by a stepwise selection. The coefficient estimates, the standard errors and the p-values ($p<0.05$) for each model are shown in Table 4, where we scaled the input variables in the [0.1,0.9] range in order to compare LLR, PLR and LRPU models. The optimal structure for the LRPU was given by 4:5:1 with 4 nodes in the input layer (T, CA, AA and pH), 5 nodes in the hidden layer (represented in Table 4) and 1 output layer (growth/no growth).

The classifiers performance of the three models tested are presented in Table 5. Of the three models tested, the LLR model correctly predicted only 80% of the cases (FC), combining all data sets, whereas the PLR model achieved a more refined fit to the data (90% of the cases were correctly classified), but the best fit was obtained using the LRPU model. Using the combined training and validation data, the accuracy of the last model was good at FC = 92%, FAR = 9.58% and POD = 93.31%, indicating good reliability in the identification of critical cases (i.e., growth limiting). Out of the 539 growth/no growth cases, the number of cases misclassified by the models ($N_{FP}+N_{FN}$) were 108 for the LLR model, 54 for the PLR model and 43 for the LRPU model. These results are in line with those obtained by Hajmeer and Basheer (2003a), who found that the logistic regression-based classifiers were less accurate than the PNN or the FEBANN-based classifiers. They carried out hybrid approaches that integrate ANNs and statistical Bayesian conditional probability estimation, or the use of PNNs in comparison with linear and non-linear logistic regression models (Hajmeer and Basheer, 2003b). They observed that these new approaches outperformed the linear and non-linear logistic regression models both in terms of classification accuracy and ease. For instance, with the hybrid ANN-Bayesian approach, they obtained a FC value of 93.9%. The non-linear logistic model (FC = 90.5%) was also an improvement on the linear model (FC = 77.1%). On the other hand, there are different approaches in scientific literature to describe growth/no growth interface of microorganisms. Le Marc et al. (2002) developed a multiplicative cardinal model to study the effect of temperature, pH, and organic acid concentration on *Listeria* growth. This model was based on assuming independent effects between input variables and limited the boundary zone at conditions which growth rate decreased to 0. Although these models provide an accurate description of bacterial growth, when increasing the number of environmental factors and interactions among them, the use of LRPU models are an alternative option to consider, because the number of interactions increases exponentially and, therefore; it will be harder to model all these interactions, especially in the extreme regions of the domain of the input variables.

Furthermore, the LRPU model generated less false negative cases (N_{FN}) (i.e., no growth cases predicted while growth cases were observed) than the other models (Table 4), in the training, validation data, and also the combined data, which implies that it would be considered fail-safe compared to the LLR and PLR models. This fact is especially important in establishing new formulations for a food product which guarantee that no growth will occur.

Table 4
Statistical parameters obtained for the Linear Logistic Regression model (LLR), Polynomial Logistic Regression model (PLR) and Product Unit Neural Network Logistic Regression model (LRPU) for the growth limits of *L. monocytogenes*

Factors	LLR			PLR		
	Coefficient.	Std. error	p-value	Coefficient.	Std. Error	p-value
Intercept	—	—	—	3.266	1.475	0.027
T^*	7.065	1.174	0.000	12.547	1.927	0.000
pH*	2.591	0.432	0.000	21.340	2.962	0.000
AA*	−10.303	1.492	0.000	−11.504	3.405	0.001
CA*	−4.920	1.552	0.002	—	—	NS
(pH)*(AA)*	—	—	NS	−33.974	10.694	0.001
(pH)*(CA)*	—	—	NS	−179.455	30.642	0.000

Parameters	LRPU	Coefficient.	Std. error	p-value
1	Intercept	10.487	2.428	0.000
1	$(T^*)^{0.417}$	32.462	5.289	0.000
2	$(pH^*)^{1.653}\ (AA^*)^{-0.915}$	115.945	19.001	0.000
3	$(pH^*)^{1.777}\ (AA^*)^{-0.767}\ (CA^*)^{0.262}$	−281.699	46.242	0.000
3	$(pH^*)^{-0.111}\ (AA^*)^{0.270}\ (CA^*)^{0.089}$	−38.326	6.467	0.000
4	$(T^*)^{0.139}\ (pH^*)^{0.654}\ (AA^*)^{-3.532}\ (CA^*)^{-1.897}$	-3×10^{-4}	5.22×10^{-5}	0.000

CA* = Scaled citric acid; AA* = scaled ascorbic acid; T^* = scaled temperature, pH* = scaled pH, NS = not significant.

Table 5
Classification table obtained for the growth limits of *L. monocytogenes*

Classifications models	Training						
	N_{TN}	N_{FP}	N_{FN}	N_{TP}	FC (%)	FAR (%)	POD (%)
LR	99	35	22	149	81.3	26.11	87.13
PLR	118	16	10	161	91.47	11.94	94.15
LRPU	122	12	9	162	93.11	8.95	94.73
Validation							
LR	76	30	21	107	78.2	28.3	83.59
PLR	90	16	12	116	88.03	15.09	90.62
LRPU	95	11	11	117	90.59	10.37	91.40
Combined							
LR	175	65	43	256	79.96	27.08	85.61
PLR	208	32	22	277	89.98	13.33	92.64
LRPU	217	23	20	279	92.02	9.58	93.31

LLR = Linear Logistic Regression model; PLR = Polynomial Logistic Regression model; LRPU = Product Unit Neural Network Logistic Regression model; N_{TN} = true negative CASES; N_{FP} = false positive cases; N_{FN} = false negative cases; N_{TP} = true positive cases; FC% = fraction of correct classified cases; FAR% = false alarm rate; POD% = probability of detection.

3.2. Interpretability of the LRPU model

Since the LRPU model proposed presented the best fit to the data observed, it seems logical to evaluate whether or not that model can predict the behaviour of *L. monocytogenes* reliably in growth limiting areas. To facilitate the comprehension and the interpretability of the LRPU model, we propose a specific case as an example: $T = 4\,^{\circ}C$; CA = AA = 0.3% (w/v) and pH = 5.5. The scaled values in the rank [0.1–0.9] of the parameters in this example will be: $T^* = 0.1$; $CA^* = 0.113$; $AA^* = 0.176$; $pH^* = 0.633$. To obtain the estimated logit P term, we multiplied each term of the equation by the scaled value of the entered parameters. The logit P term will be equal to −3.606 and the probability of growth (P) can be obtained from Eq. (9) ($P = 0.026$). It should be taken into account that concentration of organic acids is introduced as u.a.c (mM) in the model instead of % w/v. In the same way, PLR and LLR can be applied.

At higher temperatures, probability of growth increased and growth was significantly reduced at low temperatures (4 and 7 °C). The model considered the collateral effect of the temperature compared with the rest of the factors so the interactions between the pH, CA and AA were considerably important. This fact can be observed because the first term of the equation is not crossed with other factors and temperature only appears in the last term to consider the joint effect of all the variables in specific boundary areas.

However, the temperature effect was increased by reducing the pH or increasing the concentration of the organic acids used. Predictions obtained with the PLR and LRPU models fixing P at 0.9; 0.5 and 0.1 are represented in Fig. 1 (CA = AA = 0.2%). Concentration of both organic acids is expressed in % (w/v) in Fig. 1–4 to facilitate an easier interpretability, although model parameters were calculated in u.a.c (mM), as stated in Section 2. The growth probability of the LRPU model was sharply defined by small changes in the levels of the factors. Geeraerd et al. (1998) stated that the pH range over *L. monocytogenes* can grow is narrower at low temperatures; thus, growth can be reduced by applying specific combinations of temperature and pH, which separately would not enable bacterial inhibition. Predictions of LRPU model shown that the combination of low temperatures and pH were particularly significant. For instance, at 7 °C and pH 5 (CA = AA = 0%) growth probability was 0.99, but by reducing the pH to 4.5, growth was almost inhibited ($P = 0.05$). At lower pH, temperature was an important factor, since, at 7 °C, probability of growth was 0.05, whereas at temperatures higher than 15 °C, predicted probability of growth was above 0.88 (pH 4.5, CA = 0 %; AA = 0 %). At 4 °C (CA = AA = 0 %), predictions of the LRPU model showed no growth ($P < 0.5$) for pHs below 4.81. These results are in concordance with other published works. George et al. (1988) studied the growth limits of *L. monocytogenes* as a function of temperature and pH and reported that minimum pH at which growth was observed at 20 and 30 °C was 4.20 and 4.43, respectively, while at 4 °C, no growth was recorded at pHs below 5.03. Tienungoon et al. (2000) evaluated the growth boundaries of *L. monocytogenes* influenced by temperature, a_w, pH and lactic acid concentration. They found that for $a_w = 0.993$, 0% lactic acid and a temperature range of 15–30 °C, growth was observed at pH 4.5, but at temperatures below 10 °C, growth occurred at pHs above 5. Le Marc et al. (2002) included a novel term to study the combined effects of temperature, pH and organic acids on the growth limits of *L. innocua*. At 15 °C, growth was recorded at pH 4.5, but at 5 °C was not observed at pH values below 5.05. The

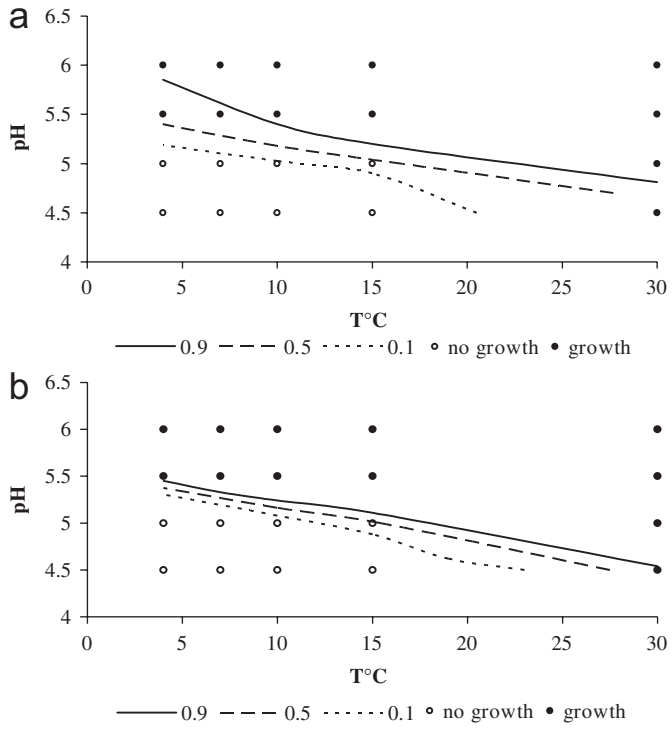

Fig. 1. Growth/no growth interfaces for the predicted growth limits of *L. monocytogenes* by the Polynomial Logistic Regression model (PLR) and the Product Unit Neural Network Logistic Regression model (LRPU) fixing the probabilities at 0.1, 0.5 and 0.9. CA = AA = 0.2%. (a) PLR model; (b) LRPU model.

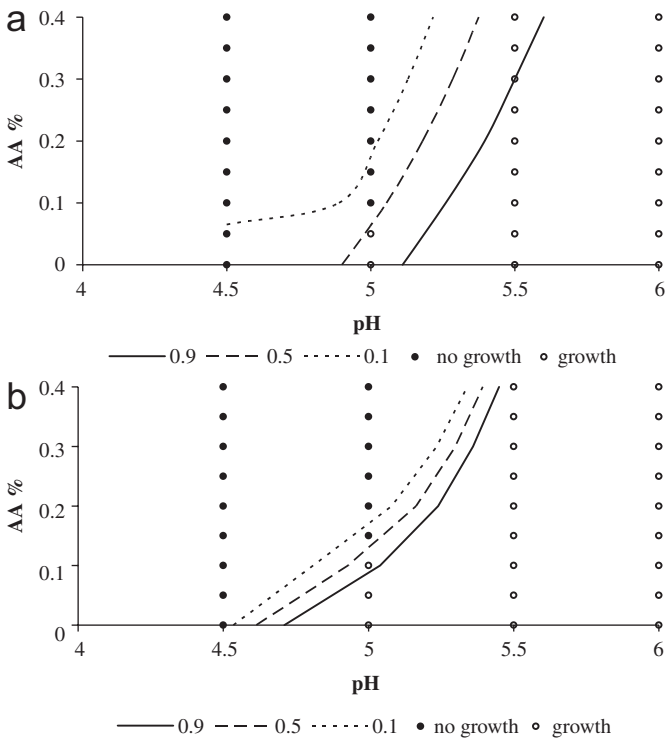

Fig. 2. Growth/no growth interfaces for the predicted growth limits of *L. monocytogenes* by the Polynomial Logistic Regression model (PLR) and the Product Unit Neural Network Logistic Regression model (LRPU) fixing the probabilities at 0.1, 0.5 and 0.9. CA = 0.2% pH 5. (a) PLR model; (b) LRPU model.

same results were also observed by Parish and Higgins (1989) and Petran and Zottola (1989). Koutsoumanis et al. (2004) have studied the growth limits of *L. monocytogenes* influenced by temperature, pH and a_w in agar. No growth was observed at 4 °C- pH 4.96; and 10 °C-pH 4.45. Similar results were observed by Cheng-An Hwang and Tamplin (2005) in mayonnaise, where *Listeria* did not grow at pH 4.5 and 5 at refrigeration temperatures (4–8 °C).

As the u.a.c. of the organic acids is highly dependent on the pH level, these interactions are expressed in the second, third and fourth term of the equation with different effects. As the acid concentration increased, there was an increase in the minimal pH that supported growth, showing an interactive effect (Le Marc et al., 2002). The second term expresses the interaction of the pH level with AA; where the pH has a greater influence on the value of this coefficient. At pH levels of over 5.4, the probability of growth was higher than 0.9 for any concentration of AA. Fig. 2 shows the relationship between these two factors (10 °C and CA = 0.2%); growth probability is higher when the concentration of AA is decreased and the pH increased.

The third and the fourth term of the equation express the interactions between CA, pH and AA. In general, the same pH effect was observed with CA and AA: the probability of growth was inversely correlated. The growth interfaces between pH and CA are shown in Fig. 3 (10 °C and AA = 0.2%); however, in comparison with AA, CA presented a greater inhibitory effect based on the u.a.c. For instance, in Fig. 4 no growth was observed at CA concentrations of 0.3% at pH 5 (10 °C and AA = 0%), whereas growth was produced at AA concentrations of 0.3% at pH 5 (10 °C and CA = 0%). These effects were incremented by reducing the pH, especially below 5. Results from Giannuzzi and Zaritzky (1996) are in agreement with the major effectiveness of citric acid when the analysis is based on u.a.c. This effectiveness is due to the physical and chemical characteristics of media as well as the chemical nature of these acids. Because of the pK values of CA ($pK_1 = 3.14$; $pK_2 = 4.77$; $pK_3 = 6.39$), it presents a strong dissociation inside microbial cells (all pK under physiological pH of cytoplasm, pH = 7.4). Results from Ahamad and Marth (1989) and Young and Foegeding (1993) agree with the inhibitory effect of citric acid and suggest that a higher concentration of net-protons is reached inside microbial cells when CA is added, in spite of passing through the cell membrane at a lower quantity. However, if the analysis is based on the undissociated fraction (u.a.f.), AA presented more inhibitory effect than CA. This fact, is due to the *pK* values of AA are far from each other ($pK_1 = 4.00$ and $pK_2 = 11.79$), so it dissociates in little proportion, comparing with CA at the same total concentration of both acids. AA is, from this point of view, the most effective acid due to its high pK_a value (Giannuzzi and Zaritzky, 1996).

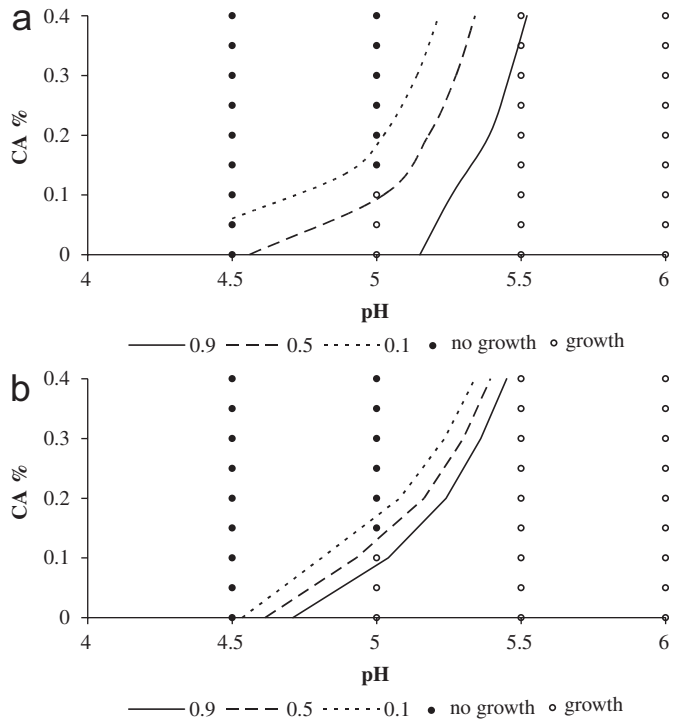

Fig. 3. Growth/no growth interfaces for the predicted growth limits of *L. monocytogenes* by the Polynomial Logistic Regression model (PLR) and the Product Unit Neural Network Logistic Regression model (LRPU) fixing the probabilities at 0.1, 0.5 and 0.9. AA = 0.2% pH 5. (a) PLR model; (b) LRPU model.

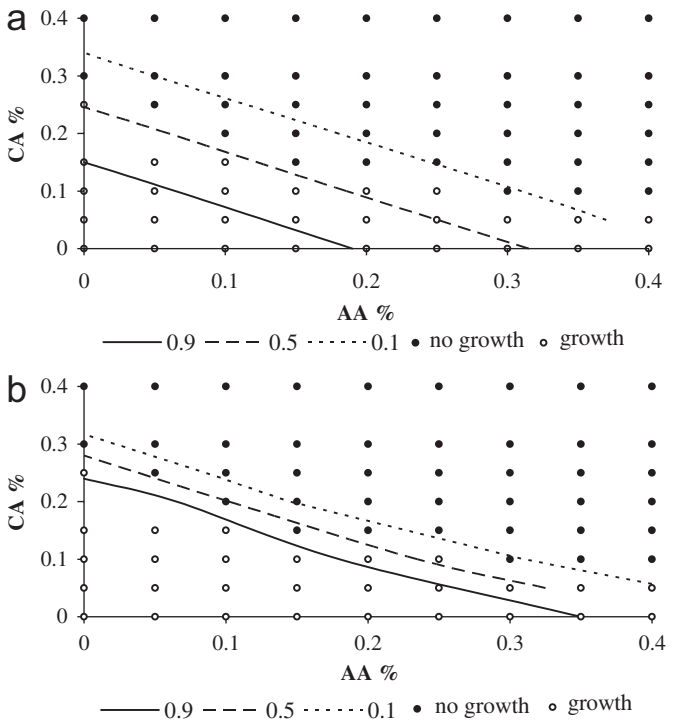

Fig. 4. Growth/no growth interfaces for the predicted growth limits of *L. monocytogenes* by the Polynomial Logistic Regression model (PLR) and the Product Unit Neural Network Logistic Regression model (LRPU) fixing the probabilities at 0.1, 0.5 and 0.9. 10 °C pH 5. (a) PLR model; (b) LRPU model.

The combined effect of all variables is represented by the last term of the equation, but no clear trends were observed, maybe because this term refers to the fit of the model to a specific boundary zone.

In summary, the pH had the greatest effect, particularly when interacting with the organic acids since small transitions produced significant changes in probability. Additionally, there was a remarkable effect on growth probability at low temperatures.

For evaluating the robustness of the LRPU model, it was submitted to a variation of the identified parameters. Predictions obtained by the model were close to data observed at stringent conditions for growth (which are more interesting to food producers). Fig. 5 represents the estimated probability as a function of temperature at different organic acids percentages at pH = 5.5. It can be seen that relatively lower concentrations of organic acids (0.25% w/v) does not produce growth inhibition even at 4 °C, so for designing a formulation of a minimally processed product, higher concentrations are needed. At concentrations above 0.3% w/v growth was inhibited at refrigeration temperatures (Fig. 5) and at 0.4% w/v *Listeria* can not growth below 10 °C. On the other hand, Fig. 6 represents the estimated probability given by the LRPU model as a function of pH at different organic acids concentrations at 7 °C. In this case, inhibition of *Listeria* growth occurred at pHs below 5.5 when concentration of citric and ascorbic acid were above 0.3% w/v. Therefore, by modifying the environmental factors, formulations of minimally processed food products can be designed in order to increase food safety regarding *L. monocytogenes* growth.

Ratkowsky (2002) noticed certain problems that arise from the use of linear and non linear logistic regression models. The use of small numbers of replicates per combination of environmental factors (due to the time-consuming nature of the experimental work) does not allow convergence to a global optimum or an appropriate set of conditions. In this case, the convergence obtained was similar for the PLR model and the LRPU model, since few replicates were used (8 per condition). A stable solution can be proposed by fixing one of the cardinal parameters (T, pH, etc.) to some realistic values. However, non-linear logistic regression, and logistic regression in general, involves binomially distributed error, and so techniques for ensuring convergence are still in progress.

In this study, we compared and assessed three different models for classifying the growth/no growth boundaries of *L. monocytogenes*. It can be noted in all cases that the predictions obtained by the LRPU model fit better to the data observed than those obtained with the PLR model and produced greater classification accuracy of the generalization data, without increasing the number of coefficients of the model.

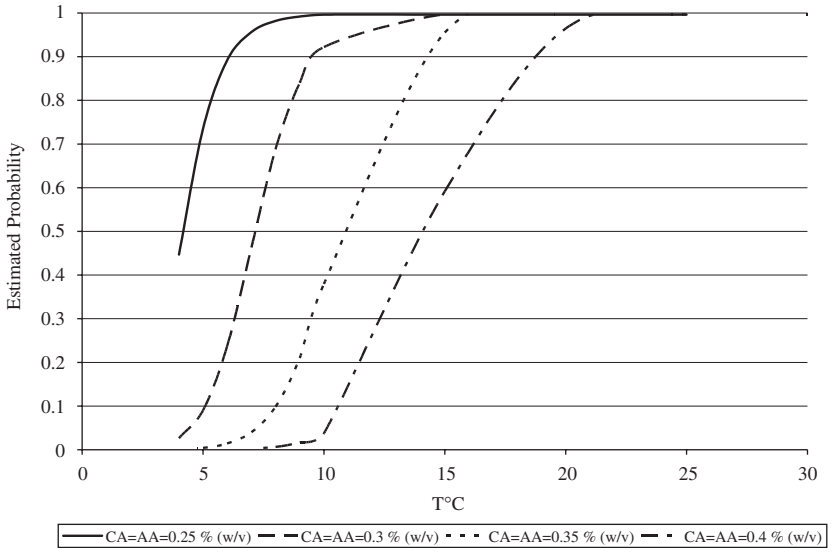

Fig. 5. Estimated probability for the LRPU model as a function of temperature at various concentrations of organic acids. pH was fixed a 5.5.

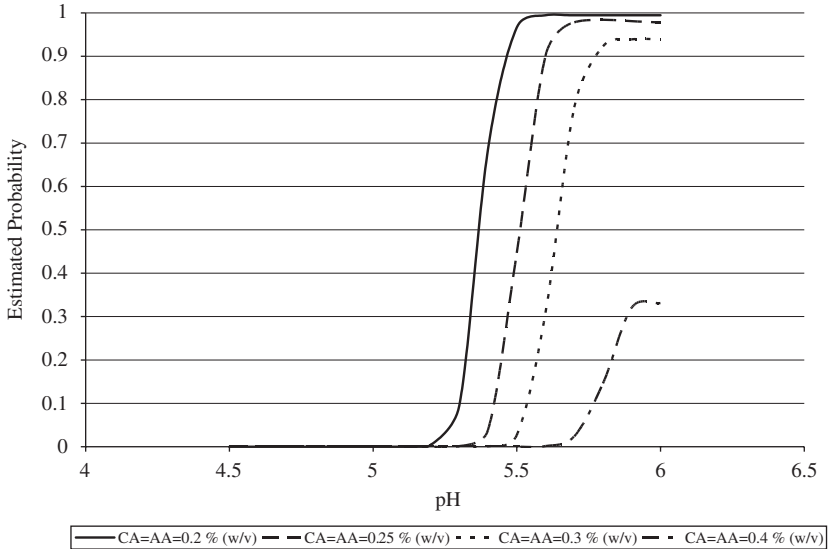

Fig. 6. Estimated probability for the LRPU model as a function of pH at various concentrations of organic acids. Temperature was fixed a 7 °C.

In conclusion, the use of LRPU models to determine growth probability under a set of conditions could constitute a valuable alternative method for mathematical modeling.

Acknowledgments

This work was partly financed by a grant provided by the Ministery of Education and Science (MEC), AGL-05-0119 (Department of Food Science and Technology, University of Cordoba) and TIN2005-08386-C05-02 (Department of Computer Science, University of Cordoba) Projects. FEDER also provided additional funding.

References

Ahamad, N., Marth, E.H., 1989. Behaviour of *Listeria monocytogenes* at 7, 13, 21 and 35 °C in tryptose broth acidified with acetic, citric or lactic acid. J. Food Prot. 52, 688–695.

Cheng-An Hwang, Tamplin, M.L., 2005. The influence of mayonnaise pH and storage temperature on the growth of *Listeria monocytogenes* in seafood salad. Int. J. Food Microbiol. 102, 277–285.

Carrasco, E., García-Gimeno, R.M., Seselovsky, R., Valero, A., Pérez, F., Zurera, G., Todd, E., 2006. Predictive model of *Listeria monocytogenes*' growth rate under different temperatures and acids. Food Sci. Technol. Int. 12, 47–57.

Durbin, R., Rumelhart, D., 1989. Products-units: a computationally powerful and biological plausible extension to backpropagation networks. Neural Comput 1, 133–142.

George, S.M., Lund, B.M., Brocklehurst, T.F., 1988. The effect of pH and temperature on the initiation of growth of *Listeria monocytogenes*. Lett. Appl. Microbiol. 6, 153–156.

Geeraerd, A.H., Herremans, C.H.M.L., Herremans, M.L., Cenes, C., Van Impe, J.F., 1998. Application of artificial neural networks as a non linear technique to describe bacterial growth in chilled food products. Int. J. Food Microbiol. 44, 49–68.

Giannuzzi, L., Zaritzky, N.E., 1996. Effect of ascorbic acid in comparison to citric and lactic acid on *Listeria monocytogenes*. Inhibition at refrigeration temperatures. Lebensmittel-Wissensch. Technol. 29, 278–285.

Gurney, K.N., 1992. Training nets of hardware realizable sigma-pi units. Neural Networks 5, 289–303.

Hajmeer, M.N., Basheer, I.A., 2003b. A hybrid Bayesian–neural network approach for probabilistic modeling of bacterial growth/no-growth interface. Int. J. Food Microbiol. 82, 233–243.

Hajmeer, M.N., Basheer, I.A., 2003a. Comparison of logistic regression and neural network-based classifiers for bacterial growth. Food Microbiol 20, 43–55.

Hervás-Martínez, C., Martínez-Estudillo, F., 2006. Logistic regresión using covariates obtained by product-unit neural network models. Pattern Recognition. Published on-line.

Hervás-Martínez, C., García-Gimeno, R.M., Martínez-Estudillo, A.C., Martínez-Estudillo, F.J., Zurera-Cosano, G., 2006. Improving microbial growth prediction by Product Unit Neural Networks. J. Food Sci. 71 (2), 31–38.

International Commission on Microbial Specifications for Foods (ICMSF), 1980. Microbial Ecology of Foods, vol. 1. Academic Press, New York.

Kirkpatric, S., Gellat, C.D.J., Vechi, M.P., 1983. Optimization by simulated annealing. Science 220, 671–680.

Koutsoumanis, K.P., Kendall, P.A., Sofos, J.N., 2004. A comparative study on growth limits of *Listeria monocytogenes* as affected by temperature, pH and a_w when grown in suspension or on a solid surface. Food Microbiol 21, 415–422.

Lanciotti, R., Sinigaglia, M., Gardini, F., Vannini, L., Guerzoni, M.E., 2001. Growth/ no growth interfaces of *Bacillus cereus*, *Staphylococcus aureus* and *Salmonella enteritidis* in model systems based on water activity, pH, temperature and ethanol concentration. Food Microbiol 18, 658–659.

Le Marc, Y., Huchet, V., Bourgeois, C.M., Guyonnet, J.P., Mafart, P., Thuault, D., 2002. Modelling the growth kinetics of *Listeria* as a function of temperature, pH and organic acid concentration. Int. J. Food Microbiol. 73, 219–237.

Martínez-Estudillo, A.C., Martínez-Estudillo, F.J., Hervás-Martínez, C., García-Pedrajas, N., 2006. Evolutionary product unit based neural networks for Regression. Neural Networks 19, 477–486.

McKellar, R.C., Lu, X., Delaquis, P.J., 2002. A probability model describing the interface between survival and death of *Escherichia coli O157:H7* in a mayonnaise model system. Food Microbiol 19, 235–247.

McMeekin, T.A., Presser, K., Ratkowsky, D.A., Ross, T., Salter, M., Tienungoon, S., 2000. Quantifying the hurdle concept by modelling the bacterial growth/no growth interface. Int. J. Food Microbiol 55, 93–98.

Parish, M.E., Higgins, D.P., 1989. Survival of *Listeria monocytogenes* in low pH model broth systems. J. Food Prot: 52, 144–147.

Petran, R.L., Zottola, E.A., 1989. A study of factors affecting growth and recovery of *Listeria monocytogenes* Scott A. J. Food Sci. 54, 458–460.

Ratkowsky, D.A., 2002. Some examples of, and some problems with, the use of nonlinear logistic regression in predictive food microbiology. Int. J. Food Microbiol. 73, 119–125.

Ratkowsky, D.A., Ross, T., 1995. Modelling the bacterial growth/no growth interface. Lett. Appl. Microbiol. 20, 29–33.

Salter, M.A., Ratkowsky, D.A., Ross, T., McMeekin, T.A., 2000. Modelling the combined temperature and salt (NaCl) limits for growth of a pathogenic *Escherichia coli* strain using nonlinear logistic regression. Int. J. Food Microbiol. 61, 159–167.

Schaffner, D.W., Labuza, T.P., 1997. Predictive microbiology where are we and where are we going? Food Technol 51, 95–99.

SPSS, I., 2003. SPSS 11.0 Advanced Models. SPSS Inc., Chicago, IL.

Tienungoon, S., Ratkowsky, D.A., McMeekin, T.A., Ross, T., 2000. Growth limits of *Listeria monocytogenes* as a function of temperature, pH, NaCl, and lactic acid. Appl. Environment Microbiol. 66, 4979–4987.

Young, K.M., Foegeding, P.M., 1993. Acetic, lactic and citric acids and pH inhibition of *Listeria monocytogenes* Scott A and the effect on intracellular pH. J. Appl. Bacteriol. 74, 515–520.