Available online at www.sciencedirect.com
ScienceDirect
Neurocomputing 72 (2008) 548–561
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Evolutionary product-unit neural networks classifiers

F.J. Martínez-Estudillo[a,*], C. Hervás-Martínez[b], P.A. Gutiérrez[b], A.C. Martínez-Estudillo[a]

[a]Department of Management and Quantitative Methods, ETEA, Escritor Castilla Aguayo 4, 14005 Córdoba, Spain
[b]Department of Computer Science and Numerical Analysis, University of Córdoba, Campus de Rabanales, 14071 Córdoba, Spain

Received 13 April 2007; received in revised form 26 October 2007; accepted 25 November 2007
Communicated by G.-B. Huang
Available online 15 January 2008

Abstract

This paper proposes a classification method based on a special class of feed-forward neural network, namely product-unit neural networks. Product-units are based on multiplicative nodes instead of additive ones, where the nonlinear basis functions express the possible strong interactions between variables. We apply an evolutionary algorithm to determine the basic structure of the product-unit model and to estimate the coefficients of the model. We use softmax transformation as the decision rule and the cross-entropy error function because of its probabilistic interpretation. The approach can be seen as nonlinear multinomial logistic regression where the parameters are estimated using evolutionary computation. The empirical and specific multiple comparison statistical test results, carried out over several benchmark data sets and a complex real microbial *Listeria* growth/no growth problem, show that the proposed model is promising in terms of its classification accuracy and the number of the model coefficients, yielding a state-of-the-art performance.

Keywords: Classification; Product-unit neural networks; Evolutionary neural networks

1. Introduction

The simplest method for the classification of patterns provides the class level given their observations via linear functions in the predictor variables. This process of model fitting is quite stable, resulting in low variance but a potentially high bias. Frequently, in a real-problem of classification, we cannot make the stringent assumption of additive and purely linear effects of the variables. A traditional technique to overcome these difficulties is augmenting/replacing the input vector with new variables, the basis functions, which are transformations of the input variables, and then using linear models in this new space of derived input features. One first approach is to augment the inputs with polynomial terms to achieve higher-order Taylor expansions, for example, with quadratic terms and multiplicative interactions. Once the number and the structure of the basis functions have been determined, the models are linear in these new variables and their fitting is a standard procedure. Methods like sigmoidal feed-forward neural networks [6], projection pursuit learning [23], generalized additive models [31], and PolyMARS [43], a hybrid of multivariate adaptive splines (multiadaptive regression splines, MARS) [22], specifically designed to handle classification problems, can be seen as different basis function models. The major drawback of these approaches is stating the optimal number and typology of corresponding basis functions.

We tackle this problem proposing a nonlinear model along with an evolutionary algorithm (EA) that finds the optimal structure of the model and estimates its corresponding coefficients. Concretely, our approach tries to overcome the nonlinear effects of the input variables by means of a model based on nonlinear basis functions constructed with the product of the inputs raised to arbitrary powers. These basis functions express possible strong interactions between the variables, where the exponents may even take on real values and are suitable for automatic adjustment. The model proposed

*Corresponding author.
E-mail addresses: fjmestud@etea.com (F.J. Martínez-Estudillo), chervas@uco.es (C. Hervás-Martínez), i02gupep@uco.es (P.A. Gutiérrez), acme@etea.com (A.C. Martínez-Estudillo).

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2007.11.019

corresponds to a special class of feed-forward neural network, namely product-unit based neural networks (PUNN) introduced by Durbin and Rumelhart [16]. They are an alternative to sigmoidal neural networks and are based on multiplicative nodes instead of additive ones. Unfortunately, the error surface associated with PUNNs is extremely convoluted with numerous local optima and plateaus. The main reason for this difficulty is that small changes in the exponents can cause large changes in the total error surface. Because of this, their training is more difficult than the training of standard sigmoidal-based networks. For example, it is well known [7] that back-propagation is not efficient in the training of product-units. Section 3 will briefly show the most relevant techniques that have been used so far to apply learning methods to product-unit networks.

On the other hand, the evolutionary approach is used to optimize both the weights and the architecture of the network simultaneously. In general, classical neural networks training algorithms assume a fixed architecture; nevertheless, it is very difficult to know beforehand what the most suitable structure of the network for a given problem will be. There have been many attempts to design the architecture automatically, such as constructive and pruning algorithms [51,56]. We used EAs to design a nearly optimal neural network architecture because better results have been obtained using this heuristic [4,62]. This fact, together with the complexity of the error surface associated with a PUNN, justifies the use of an EA to design the topology of the network and to train its corresponding weights. The evolutionary process determines the number of basis functions, associated coefficients and corresponding exponents in the model.

We use the softmax activation function and the cross-entropy error function [6]. Therefore, from a statistical point of view, the approach can be seen as a nonlinear multinomial logistic regression [30], where we optimize log-likelihood using evolutionary computation. Actually, we attempt to estimate conditional class probabilities using a multilogistic model with the nonlinear model given by PUNNs.

We evaluate the performance of our methodology on seven data sets taken from the UCI repository [7], and on a real microbial growth/no growth problem in order to determine the growth limits of *Listeria monocytogenes* [2,18] to assure microbial safety and quality in foods. Empirical and statistical test results show that the proposed method performs well when compared to several other learning classification techniques. We obtain a classifier with interesting results in terms of classification accuracy and number of hidden nodes. Moreover, we show graphically the classification task carried out by the product-unit model together with its capability to both capture the interactions between the variables and to reduce the dimension of the input space. This reduction of dimensionality facilitates the study of the behavior of corresponding basis functions and the relevance of each input variable in the final model. This paper is organized as follows: Section 2 shows the main related works; Section 3 is devoted to a description of PUNNs; Section 4 describes the evolution of PUNNs; Section 5 explains the experiments and the comparison test carried out; and finally, Section 6 summarizes the conclusions of our work.

2. Related works

We start by giving a brief overview of the different methods that use basis functions to move beyond linearity. The first method cited to solve classification problems is conventional statistical discriminant analysis [30], which assumes that the measurement vectors in each class follow a normal multivariate distribution. If the covariance matrices of the measurements in each class are the same, the method shows that the regions created by Bayes' decision rule are separated by boundaries, which are linear in the input variables. When the conventional assumption of the equality of covariate matrices is dropped, Bayes' decision rule gives quadratic boundaries. In many examples, the inadequacy of linear or quadratic discriminant analysis for the purpose of classification made it necessary to look for approaches that could approximate highly nonlinear class boundaries. Instead of assuming specific distributions for the inputs and using them to calculate conditional class probabilities, one can estimate these classes directly from training sample cases.

A number of methods based on nonparametric regression [30], which are capable of approximating highly nonlinear class boundaries in classification problems, have been developed in the last few years.

Generalized additive models [31] comprise automatic and flexible statistical methods that may be used to identify and characterize nonlinear effects. The generalized additive model approximates multidimensional functions as a sum of univariate curves. Univariate functions are estimated in a flexible manner, using an algorithm whose basic building block is a scatter plot smoother, for example, the cubic smoothing spline. The additive model manages to retain interpretability by restricting nonlinear effects in the predictors in order to enter them into the model independently of each other. Generalized additive models provide a natural first approach to relaxing strong linear assumptions.

Bose [8] presented a method, classification using splines (CUS), somewhat similar to the neural network method, which uses additive cubic splines to estimate conditional class probabilities. Afterwards, the same author presented a modification of CUS, named "the method of successive projections", to solve more complex classification problems [9]. Although this method was presented using CUS, it is possible to replace CUS by any nonparametric regression-based classifier.

Kooperberg et al. [43] propose an automatic procedure that uses linear splines and their tensor products. This method is a hybrid of the MARS [22] called PolyMars, specifically designed to handle classification problems. It